Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный институт культуры»

На правах рукописи

СУЛЕЙМАНОВ Руслан Сулейманович

ИНТЕГРАЦИЯ ЦИФРОВЫХ ИНФОРМАЦИОННЫХ РЕСУРСОВ В ЭЛЕКТРОННЫЕ БИБЛИОТЕКИ

Специальность 05.25.05 – Информационные системы и процессы

Диссертация
на соискание ученой степени кандидата технических наук

Научный руководитель**:**
кандидат технических наук, доцент
ГОНЧАРОВ Михаил Владимирович

Москва - 2020

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Актуальность темы. Развитие науки и образования в XXI веке невозможно без развития информационных технологий, в частности сегодня, в эпоху нарастающей цифровизации, необходим принципиально новый подход к разработке информационного обеспечения образования и науки. Быстрое развитие информационных технологий дало возможность обеспечения требуемой генерации и распространения научной и образовательной информации. Традиционно функцию накопления, использования и передачи знаний исполняли библиотеки: научные, общедоступные, специализированные. Международная федерация библиотечных ассоциаций и учреждений сообщает, что в мире сегодня существует более 569,6 тысяч традиционных библиотек [48], в том числе более 100 тысяч только в Российской Федерации. Отметим, что с конца 1990-х годов наряду с традиционными в практику информационного обслуживания пользователей начали входить электронные библиотеки. Одним из способов формирования и передачи знаний, обеспечивающих удобство и простоту получения информации, сегодня являются электронные библиотеки, число которых растет как в России, так и во всем мире [16].

На данный момент в стране создаются и используются электронные библиотеки в разных сферах деятельности, и большую роль в этом играют библиотеки и информационные центры страны, научно-исследовательские институты и образовательные учреждения [7]. Нельзя не отметить две наиболее крупные и известные электронные библиотеки национального масштаба: «Президентскую библиотеку имени Б. Н. Ельцина» (Санкт-Петербург) и «Национальную электронную библиотеку», оператором которой является Российская государственная библиотека. Тем не менее, несмотря на богатство электронным контентом этих двух национальных систем и существующих электронных библиотек в библиотеках, институтах и вузах страны, потребности науки, образования и культуры существенно шире. На данный момент имеется

потребность, скорее даже уже требование, ученых, специалистов и обучающихся в развитии информационного обеспечения, прежде всего в расширении доступа к большому числу универсальных и профильных электронных ресурсов. Современная наука требует большого охвата разных отраслей с точки зрения создания цифровых коллекций и отдельных универсальных или проблемно-ориентированных электронных библиотек. С учетом имеющихся современных средств и инструментария необходимо учитывать не только потребности ученых, исследователей, преподавателей и студентов, но и возможности современной научной коммуникации, позволяющей оптимизировать создание, эффективное распространение и использования электронных коллекций (библиотек).

Одной из основополагающих проблем при создании электронной библиотеки является интеграция данных [30, 39, 37], так как их объем постоянно увеличивается, что приводит к тому, что становится все сложнее их интегрировать с учетом не только объема, но и форматов представления и, главным образом, обеспечения необходимой релевантности. Во-первых, требуется обеспечить непрерывный и удобный доступ для получения контента. Во-вторых, и это главное, необходимо извлечь метаданные, которые содержатся внутри документа и однозначно определяют его. При этом если сами исходные документы хранятся в одинаковых форматах, то в таком случае можно разработать правила и соответствующее программное обеспечение, позволяющие анализировать эти данные, либо применить готовые созданные парсеры данных [15]. Однако в случае если данные хранятся в разных форматах, не обойтись без написания собственного парсера. В качестве альтернативного метода может выступать создание универсального конструктора правил интеграции полей [72]. Сам конструктор может быть реализован в виде веб-интерфейса, позволяющего извлекать необходимые целевые поля из документов или страниц внешней электронной библиотеки. Эта задача актуальна именно сегодня, так как позволяет повысить эффективность интеграции данных и обеспечить работу с любыми типами источников.

Одним из основных критериев удобства пользования электронной библиотекой является возможность быстро найти искомый документ [49], что обеспечивается поиском по метаданным. Тем не менее иногда необходимые документы публикуются в разных коллекциях, в том числе в виде файлов на диске, не сопровождаемых достаточным набором метаданных, необходимым для релевантного поиска, что создает пользователю большие проблемы при поиске документа. Однако при этом само содержимое документов может включать нужные данные: название документа, фамилию автора, информацию об издательстве и так далее. В данном исследовании рассматривается способ извлечения метаданных из полных текстов документов для повышения уровня их идентификации в электронной библиотеке.

Таким образом, актуальность данного исследования обоснована необходимостью проектирования электронных библиотек с учетом разнородности и распределенности представления данных и обеспечения требований, предъявляемых к ресурсам Интернета: прежде всего быстроты отклика на запрос, интуитивно понятного и удобного в использовании интерфейса, а также возможности интеграции ресурсов из максимального количества источников на основании использования метаданных [8].

Степень научной разработанности разных аспектов темы исследования достаточно высока. В отечественной и зарубежной библиотечно-информационной науке в последние годы подготовлено немало статей и обзоров по различным вопросам, вошедшим в рамки изучаемой в исследовании проблемы.

В своей работе автор опирался на методологии проектирования электронных библиотек, введенные Антопольским А.Б., Земсковым А.И., Шрайбергом Я.Л. Вопросы, связанные с организацией работы поисковых библиотечных систем были затронуты в работах Каленова Н.Е., Колосова К.А., Соколинского К.Е., Сотникова А.Н. Проблемы интеграции информации из распределенных источников описаны в трудах Погорелко К.П., Рябова В.И., Серебрякова В.А., Соболевской И.Н.

Проблемы интеграции существующих библиотечных ресурсов в единую базу данных решаются в таких проектах как “Научное наследие России”.

Научные труды многих известных ученых позволили определить цель исследования, однако анализ имеющейся литературы показал недостаточную степень изученности проблемы интеграции информации из распределенных источников с применением методики извлечения метаданных из полнотекстовых электронных документов, что является одним из наиболее значимых аргументов для подготовки настоящего диссертационного исследования.

Цель и задачи исследований. Целью диссертационной работы является улучшение качества интеграции цифровых информационных ресурсов из разных источников с помощью модели и методики, учитывающих разные структуры данных.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Провести анализ существующих способов и механизмов интеграции данных в электронных библиотеках.
2. Разработать модель и спроектировать эффективный конструктор правил интеграции информации из распределённых источников (базы данных, веб-сайты и полнотекстовые документы в формате PDF).
3. Разработать методику извлечения метаданных из полных текстов оцифрованных документов, что позволит повысить полноту предоставленных метаданных текстов на естественном языке.

В качестве **теоретической и методологической** основы диссертации выступают исследования и разработки отечественных и иностранных ученых в области построения баз данных, интеграции материалов библиотек, извлечения метаданных.

При работе над диссертацией автором были использованы труды российских и зарубежных ученых Антопольского А. Б., Вислого А. И., Гончарова М. В., Земскова А. И., Калёнова Н. Е., Колосова К. А., Лопатиной Н. В., Лютецкого В. М.,

Мазурицкого А. М., Соколинского К. Е., Сотникова А. Н., Тютюнника В. М., Цветковой В. А., Шрайберга Я. Л., Tillett B. B. и других.

В работе использованы **методы** структурного анализа, системного анализа теории проектирования баз данных, теории объектно-ориентированного программирования, теории анализа текстов на естественном языке.

Программное обеспечение для прогностической части работы реализовано средствами языка PHP в связке с СУБД MySQL, поисковой машины Sphinx и Яндекс «Томита-парсер».

Научная новизна работы состоит в обосновании и разработке новой экспериментальной методики интеграции цифровых данных из разнородных и распределённых источников для электронных библиотек. Методика позволяет выявить качественно новые закономерности представления метаданных в цифровых документах, являющихся единицей записи данных из коллекций в электронных библиотеках:

1. Разработана и обоснована модель конструктора правил интеграции информации из распределенных источников для электронных библиотек, позволяющая упростить процесс наполнения базы данных электронных документов и доказавшая перспективность для использования в построении электронных библиотек.
2. Разработана и обоснована методика извлечения метаданных, в том числе новые грамматики и словари на основе естественного языка, используемая для анализа полнотекстовых оцифрованных документов, а также программная реализация механизма извлечения метаданных из полнотекстовых документов.

Теоретическая значимость

Выявлены проблемы интеграции информации из распределенных источников, возникающие в основном из-за разных форматов хранения метаданных.

Создана и обоснована модель «Конструктора правил интеграции электронных документов из распределенных источников для электронных

библиотек». Теоретическая значимость данной модели заключается в расширении представлений о механизмах формирования электронных библиотек, которая, в том числе, раскрывает особенности построения справочно-поискового аппарата электронных библиотек.

Применительно к проблематике диссертации результативно использована методика извлечения метаданных, применяемая для анализа полнотекстовых оцифрованных документов.

Практическая значимость и реализация результатов работы

Теоретические и экспериментальные результаты, полученные в ходе диссертационного исследования, прошли апробацию и были внедрены в Московском педагогическом государственном университете. Разработанные методики используются в управлении фондом электронной библиотеки Московского педагогического государственного университета. Отдельные модули автоматизированной системы управления электронной библиотекой и модуля интеграции данных используются в управлении библиотечным фондом Московского городского педагогического университета.

В открытый репозиторий по лицензии GNU General Public License (универсальная общественная лицензия GNU) выложен исходный код конструктора правил интеграции информации из распределенных источников, который позволяет автоматизировать сбор и обработку данных и метаданных оцифрованных печатных документов, в том числе книг.

Разработанный конструктор позволил объединить имеющиеся оцифрованные материалы для электронной библиотеки Московского педагогического государственного университета и автоматизировать управление фондом библиотеки Московского городского педагогического университета в части наполнения электронной библиотеки метаданными.

Результаты диссертационного исследования были использованы в управлении фондом библиотеки Московского педагогического государственного университета, что подтверждается наличием справки о внедрении.

По результатам диссертационного исследования были зарегистрированы две программы для ЭВМ: № 2012619529 - «Система управления контентом электронной библиотеки», дата регистрации 22.10.2012 (совместно с Шабановым Б.М., вклад автора диссертации - постановка задачи); № 2019661660 - «Конструктор правил интеграции данных для электронных библиотек», дата регистрации 05.09.2019 (без соавторов).

Положения, выносимые на защиту:

1. Анализ имеющихся способов интеграции информации из распределённых источников выявил проблемы, возникающие в основном из-за разных форматов хранения метаданных в электронных библиотеках.
2. Для решения проблем интеграции информации из распределённых источников в разных форматах хранения метаданных разработана и обоснована модель конструктора правил интеграции информации из распределённых источников для электронных библиотек, позволяющая упростить процесс наполнения базы данных электронных документов. Модель была апробирована и доказала перспективность для использования в построении электронных библиотек.
3. Для повышения полноты предоставленных метаданных в электронных библиотеках разработана и обоснована методика извлечения метаданных, в том числе новые грамматики и словари на основе естественного языка, используемая для анализа полнотекстовых оцифрованных документов, а также программная реализация механизма извлечения метаданных из полнотекстовых документов.

Достоверность полученных научных результатов подтверждена результатами практических применений, положительными результатами их обсуждения на научных конференциях.

Апробация работы. Основные положения работы докладывались на XI научно-практической конференции «Современные информационные технологии в управлении и образовании» (Москва, 2012); XVII научно-практическом семинаре

«Информационное обеспечение науки: новые технологии» (Таруса, 2013); III международной научно-практической конференции Innovative Information Technologies (Прага, 2014); на Московском международном салоне образования в 2018 и 2019 годах.

Личный вклад. Автором самостоятельно поставлены цель и задачи работы, разработана структура базы данных электронной библиотеки, позволяющая интегрировать информацию из разных источников, разработан конструктор полей интеграции данных, разработан метод извлечения метаданных из полнотекстовых документов, разработана программа эксперимента, проведен анализ результатов эксперимента и выявлены основные закономерности извлечения метаданных.

Результаты научного исследования отражены в семи публикациях, большая часть публикаций сделана лично соискателем, в том числе две статьи в журналах, рекомендуемых ВАК для публикации результатов диссертаций на соискание ученой степени кандидата технических наук по специальности 05.25.05.

Объем и структура диссертации. Текст диссертационной работы состоит из введения, двух глав, основных выводов по каждой главе, заключения, списка литературы и приложений. Диссертация содержит 131 страницу машинописного текста, 12 рисунков и 3 таблицы. Библиография включает 96 наименований.

ГЛАВА 1. СОВРЕМЕННЫЕ ПОДХОДЫ К ИНТЕГРАЦИИ ДАННЫХ В ЭЛЕКТРОННЫХ БИБЛИОТЕКАХ

Электронная библиотека, согласно определению, введенному А. Б. Антопольским [7], представляет собой информационную систему, позволяющую надежно хранить и эффективно использовать различные доступные коллекции электронных документов разного вида (текстовых, графических, мультимедийных и других). При этом документы могут быть размещены как в самой системе, так и доступны ей через интернет или интранет [20, 21]. Возможность создания электронных библиотек была обусловлена развитием современных технологий, такими как электронные архивы, интернет, распространение выпуска электронных изданий. Появление электронных архивов обеспечило опыт массовой оцифровки бумажных документов, систематизации и хранения больших объемов электронных документов, а также привело к развитию корпоративных баз данных [81] с документами, которые стали первым шагом к появлению корпоративных электронных библиотек [44].

Развитие интернета привело к целому ряду последствий, связанных с электронными библиотеками: созданию новой системы поиска, в частности поиска по тексту документа, возможности самостоятельной публикации материалов, созданию различных веб-порталов.

Развитие выпуска электронных изданий привело к развитию культуры электронных публикаций, например, публикации научной информации [15] в электронных рецензируемых журналах. Книжные издания также в большинстве случаев имеют электронную копию. Для обеспечения защиты авторских прав [26] появились технологии защиты электронных книг от несанкционированного копирования и использования.

Первыми шагами к появлению электронных библиотек стало создание электронных библиотечных каталогов, обеспечивающих деятельность библиотеки по каталогизации, заказу, поиску и книговыдаче. Изначально электронные библиотечные каталоги хранились в едином файле, который представлял собой таблицу, и благодаря этому было возможно искать сведения о материалах в едином месте. По мере появления баз данных стало возможным разделять все записи на отдельные поля (метаданные), такие как наименование, сведения об авторе, издательстве и другие. Подобное разделение позволило существенно упростить поиск материалов по каталогу. Примерно тогда же публиковались первые сетевые системы [3], позволяющие обеспечивать удаленный доступ к каталогу материалов. Например, система доступа к электронному библиотечному каталогу была запущена в США [12] в 1975 г. (в университете штата Огайо) и в 1978 г. (в Публичной библиотеке г. Далласа).

Первая библиотека, которая изначально была создана как электронная, появилась в 1966 году (Education Resources Information Center) при поддержке Министерства образования США. На данный момент в ней содержится 1,5 миллиона записей.

В данной главе более подробно рассмотрены предпосылки создания электронных библиотек, а также существующие форматы и стандарты метаданных, на основании которых происходит интеграция ресурсов библиотек.

1.1 Информационные массивы и электронные библиотеки

Современный этап развития общества характеризуется увеличением роли информации и созданием глобального информационного пространства [32], обеспечивающего быстрый доступ широких слоев населения к знаниям [75]. Количество информации в мире растет экспоненциально, в том числе за счет того, что информация стала одним из главных ресурсов, наряду с энергетическими, сырьевыми, финансовыми и другими. На протяжении веков информация

генерировалась исключительно человеком в устном или печатном виде, однако с развитием технологий информация накапливается также путем, например, сохранения компаниями данных о покупателях, операциях, а также хранения информации, генерируемой сенсорами, которые встроены в мобильные телефоны, автомобили, системы безопасности [39] и т. д. Увеличение количества информации связано и с появлением интернета и социальных сетей [35, 67].

Для оценки объемов информации можно осуществить мониторинг объемов трафика. Согласно данным CISCO [90], в 2020 году объем мирового интернет-трафика достигнет 161,3 экзабайта в месяц, что почти в 3 раза больше, чем в 2015 году (53,2 экзабайта). В условиях увеличивающегося объема информации остро стоит вопрос организации данных для максимальной эффективности их восприятия [6]. По оценкам экспертов, до 2020 года количество данных будет увеличиваться как минимум вдвое каждые 2 года [95]. Согласно исследованию компании Digital Universe, в ближайшие 5 лет объем данных на планете вырастет до 40 зеттабайт, то есть к 2021 году на каждого живущего на Земле человека будет приходиться более 5 террабайт [91, 96].

Стремительное развитие информационных технологий провоцирует постоянное увеличение объемов создаваемой информации в интернете [86]. С ростом количества новой информации растут и потребности в достоверных и качественных данных. Стоит заметить, что само по себе увеличение объемов данных не приводит к улучшению их качества. Информация и данные часто бывают ошибочны или нерелевантны исходным целевым запросам.

Появление традиционных библиотек существенно упростило хранение и поиск качественной и достоверной информации. Однако с ростом технологического развития современных средств коммуникации пользователям (ученым, обучающимся [41] и другим целевым группам) требуются более оперативные средства для доступа к информации в библиотеке, чем ее ручной перебор и поиск нужных фрагментов в тексте. К тому же само по себе посещение библиотеки занимает дополнительное время с учетом дороги, возможной очереди

и других факторов. Дополнительная сложность заключается в том, что особенно редкие и ценные материалы могут храниться в разных библиотеках [17], к которым у потенциального читателя может отсутствовать доступ [14, 84]. Для получения доступа требуется либо пройти регистрацию с верификацией, либо являться сотрудником определенной организации [74].

Поиск в интернете намного удобнее и зачастую проще для пользователей [36]. Для этого обычно используют крупные поисковые системы [73]. Министерство культуры Российской Федерации сообщает о том, что охват населения нашей страны библиотечным обслуживанием падает – менее 35% [60], а аудитория российского интернета наоборот растет – по данным на 2018 год в интернет выходит более 87 млн человек, что составляет около 68% населения нашей страны. Исходя из этих данных пользователям, нуждающимся в получении информации, будет проще осуществить ее поиск через интернет либо напрямую обратиться в интересующие их предметные электронные библиотеки [13].

К тому же при использовании электронной библиотеки пользователь получит требуемую информацию в уже подготовленном оцифрованном виде. Сам материал не надо будет перелистывать, переписывать и так далее. Проблема заключается в том, что данные материалы могут быть недостоверными или неполными [16]. Однако несмотря на это большинство пользователей скорее воспользуется поиском в интернете [18], а не очным посещением традиционной библиотеки, так как в данном случае он сэкономит массу времени.

С ростом объемов данных и информации появилось такое понятие как «информационный массив». Он представляет собой собрание информации, используемой как нечто единое целое [40]. В качестве информации могут рассматриваться любые материалы – книги, монографии, мультимедийные файлы и так далее.

В качестве характеристик информационных массивов выделяют следующие особенности:

1) внутри массива содержатся атомарные информационные единицы, к которым можно получить отдельный доступ;
2) собрание массива сопровождается упорядоченным сбором и систематизацией информации;
3) часто массиву свойственна тематическая однородность;
4) сам по себе массив возможно идентифицировать как автономный архив информации;
5) массив можно количественно оценить.

Чаще всего организация информационных массивов представляет собой базу данных как наиболее удобный способ доступа к накопленной информации [34].

Первым шагом к хранению информации являлись файлы и файловые системы, позволявшие хранить и изменять информацию. Однако файловая система не позволяла обрабатывать одновременно большие объемы информации нескольким пользователям сразу, что привело к созданию новой системы управления информацией – системе управления базами данных (СУБД). Первая промышленная СУБД была введена в эксплуатацию в 1968 году. Со времен появления первых баз данных происходило их развитие: начиная от иерархических и сетевых баз данных к реляционным СУБД. Вместе с развитием баз данных развивались и языки описания и модификации данных, например, SQL (один из самых широко используемых языков запросов, созданный в 1985 году), инструменты моделирования данных, индексирования и организации данных. Переход от доступа к базам данных с одного компьютера к распределенному стал возможен благодаря параллельной обработке транзакций, при которой осуществляются последовательные операции над базой данных, производимые с разных компьютеров при сохранении целостности данных. Это дало возможность организовать параллельную обработку информации при поддержке целостности базы данных, что впоследствии привело к развитию реляционных баз данных как основного типа баз данных для хранения больших информационных массивов. Во многом этому способствовало появление специальных методов обработки

транзакций OLTP (on line transaction processing). OLTP представляет собой способ организации базы данных, при котором система работает небольшими по размерам транзакциями, но идущими большим потоком, при этом клиенту требуется от системы минимальное время отклика [93].

Развитие баз данных привело к появлению новых моделей данных, таких как объектно-ориентированные, объектно-реляционные, дедуктивные модели. Развитие информационных технологий, появление персональных компьютеров стало импульсом к созданию большого количества предметно-ориентированных баз данных (разделенных по тематике или по типам материалов), а также глобальных информационных систем, таких как интернет [30].

К информационным массивам можно отнести любые базы данных, организованные для хранения и использования информации в определенных целях: интернет-ресурсы, каталоги, фонды и т. д. Одним из видов информационных массивов являются библиотечные фонды, в том числе и электронные библиотеки. Разнообразие информационных массивов привело к необходимости их описания, для чего используется система метаданных. Стандарты хранения данных в электронных библиотеках будут подробнее рассмотрены в разделе 1.2.

Основным средством передачи знаний на протяжении веков являлись книги, при этом нарастание объемов печатных изданий привело к созданию библиотек – как общего профиля, так и узкоспециализированных. Однако при переходе к информационному этапу развития общества количество производимой информации и публикаций традиционные библиотеки столкнулись с трудностью хранения больших массивов данных. Более того, доступ к печатным изданиям может быть осуществлен только лично, что является значительным неудобством в современном мире. Таким образом, все большее распространение получают электронные библиотеки, так как позволяют хранить оцифрованные печатные документы, а также материалы в других форматах (видео, изображения, звуковые файлы) [71].

Развитие информационных технологий создало благоприятные условия для создания электронных материалов, при этом широкое распространение электронных библиотек стало возможным благодаря появлению персональных компьютеров, а вслед за ними – смартфонов и планшетов [95]. Именно благодаря этим технологиям появилась возможность постоянного доступа к электронным материалам, включая аудио-, видеоматериалы, изображения, при этом качество материалов благодаря высокому разрешению может быть значительно выше, чем у печатной продукции. Более того, современные вычислительные средства позволяют открывать несколько документов сразу. В случае текстовых документов возможен (при соответствующих форматах) поиск по тексту [72], что сокращает время, необходимое для поиска нужной информации. В случае изображений, схем и т. д. компьютерные технологии позволяют увеличивать или уменьшать детали изображения, что, при должном качестве документа, позволяет изучить информацию детально. Видеоматериалы могут быть доступны в любое время при наличии доступа к электронной библиотеке. Любые форматы документов позволяют копирование информации в том или ином виде, неограниченный доступ к документам в любое время, перенос информации на другие носители.

Цифровые библиотеки остаются формой традиционных библиотек, меняются только средства доступа к контенту. Электронная библиотека по сути является информационной системой. Внутри этой системы содержатся оцифрованные материалы, доступ к которым может быть получен при помощи специального программного обеспечения с возможностью разграничения доступа по уровням, типам пользователей или иным критериям. Кроме того, электронная библиотека может содержать материалы, опубликованные на внешних ресурсах (другие электронные библиотеки, информационные архивы и т. д.), таким образом обеспечивая расширение возможностей традиционных типов библиотек. При этом отмечается, что электронные библиотеки заняли свою нишу в общей структуре передачи знаний. Дальнейшее их развитие связано с переводом все большего количества библиотечных фондов в электронный формат, а также с интеграцией

библиотек. При этом наблюдаются существенные отличия между глобальной сетью и интегрируемыми электронными библиотеками, так как электронная библиотека обладает характеристиками информационного массива и, соответственно, подразумевает упорядоченность.

В качестве характеристик электронных библиотек можно выделить следующие:

– доступ к библиотеке возможен через интернет с использованием пользовательского интерфейса;

– материалы библиотеки могут быть снабжены метаданными и могут храниться в разных каталогах и рубриках;

– сами материалы могут быть оцифрованными копиями печатных и иных материалов;

– возможность интеграции данных по какому-либо признаку.

Развитие информационных технологий и взрывной рост объема информации привел к значительной проблеме роста недостоверной информации. Недостоверная, то есть не соответствующая действительности, информация может содержать сведения о том, что никогда не существовало, или иметь ограниченные или некорректные данные.

Для оценки достоверности информации зачастую оценивается достоверность источника. Оценка достоверности не производится для художественных произведений, так как события и явления, описанные в произведениях, могут быть плодом воображения автора или интерпретатора.

При этом стоит отметить, что для научного сообщества достоверность информации представляет особую важность, так как публикации, использующие непроверенную информацию или сомнительные источники, могут подорвать авторитет автора публикации. Использование электронных библиотек позволяет пользователям быть уверенным в достоверности полученной информации, так как электронные библиотеки, аналогично традиционным библиотекам, содержат проверенные публикации.

Как и к традиционным библиотекам, к электронным библиотекам может быть применена разная классификация, однако классификация электронных библиотек более разнообразна за счет различия в возможностях преподнесения информации.

Согласно ГОСТ Р 7. 0. 96 ─ 2016. ЭЛЕКТРОННЫЕ БИБЛИОТЕКИ [2] выделяются следующие типы электронных библиотек:

По механизму создания:

1) наполняемая вручную оператором или администратором электронной библиотеки;
2) механизм объединения информации из внешних источников (агрегированный механизм);
3) смешанный механизм, использующий одновременно ручной и агрегируемые механизмы наполнения.

По способу организации:

1) встраиваемая электронная библиотека, являющаяся частью более крупного информационного ресурса;
2) отдельная, то есть организованная в виде изолированного информационного массива электронных объектов.

По способу доступа:

1) публично доступные библиотеки;
2) библиотеки, требующие регистрации перед доступом к ресурсам;
3) библиотеки, требующие подписания договора на доступ к информационным ресурсам.

По статусам:

1) мировая,
2) государственные,
3) корпоративные,
4) публичные,
5) личные.

Возможным является дополнительная классификация электронных библиотек по следующим признакам:

– инициатор создания библиотеки;

– вид(ы) литературы, представленный в библиотеке;

– форматы представляемых документов;

– организация библиотеки.

Среди инициаторов создания библиотек могут быть выделены:

– реальные библиотеки, переводящие свои фонды в электронный вид;

– государство в виде проектов по созданию электронных библиотек для расширения донесения знаний до населения;

– бюджетные учреждения (вузы [33], школы, НИИ);

– частные организации;

– физические лица.

По видам литературы библиотеки могут быть разделены на библиотеки художественной литературы и библиотеки, содержащие научную или учебную литературу [42]. Близким является разделение библиотек по содержанию, где можно выделить следующие типы:

– универсальные электронные библиотеки – содержащие документы, относящиеся к разным областям знаний;

– тематические электронные библиотеки – содержащие документы одной области знаний.

В электронных библиотеках могут быть представлены как документы одного формата, например, только текстовые документы, так и документы разных форматов, включая мультимедийные.

Электронная библиотека организационно может быть как отдельным ресурсом, так и частью другого ресурса.

Возможны и другие классификации электронных библиотек, так как не существует единой принятой всем библиотечным сообществом их классификации. При необходимости определения принадлежности электронной библиотеки к той

или иной категории происходит применение тех параметров, которые актуальны для конкретной цели.

Создание электронной библиотеки требует определения общесистемных требований и правил к разработке [1]. В качестве изначальных принципов при проектировании автоматизированной библиотечно-информационной среды [9] можно выявить: однократность ввода исходных данных в библиотеку, выявление и использования определенного принципа хранения данных, доступ через сетевые ресурсы, обеспечение совместимости со внешними системами и сервисами, например с другими электронными библиотеками.

Изначально созданием электронных библиотек начали заниматься специалисты информационных технологий, так как структура электронных библиотек аналогична структурам других информационных систем. Создание электронной библиотеки, как и создание других типов информационных систем, включает стандартные стадии от формирования требований и технического задания до ввода в эксплуатацию и сопровождения системы.

Однако разработка электронной библиотеки имеет особенности, которые нехарактерны для других информационных систем, в частности необходимость описания файлов и хранения их метаданных.

1.2 Метаданные и стандарты хранения данных в электронных библиотеках

На данный момент, несмотря на увеличивающуюся роль компьютеров в жизни общества, генерация, оценка, классификация и актуализация информации по-прежнему является деятельностью, которую осуществляет исключительно человек [72]. В основном это связано с тем, что процесс обработки человеческой речи (языка) является сложной задачей для автоматизации. Для упрощения компьютерной обработки информации, содержащейся в интернете, в 1998 году Тимом Бернерсом-Ли (создатель HTTP, WWW, URI и HTML [46]) была

предложена концепция Семантической паутины [19] – направление развития интернета, в котором данные каждой страницы должны быть представлены как на обычном языке, так и в виде метаданных, пригодных для машинной обработки.

В интернете для идентификации элементов используются специальные ссылки – «Унифицированные идентификаторы ресурсов», или сокращенно URI (Uniform Resource Identifier), которые являются основополагающим элементом Семантической паутины [92]. URI является уникальным адресом ресурса и используется для указания на конкретный ресурс или материал (сайт, изображение, документ и т. д.).

Эти адреса часто используются для определения отдельного пути в интернете к тому или иному объекту, чтобы иметь возможность обратиться к нему отдельно.

В основном в интернете данные представлены в виде текста на естественном языке. Изначально эти тексты были предназначены исключительно для обработки их человеком [52], но в последнее время появилась масса компьютерных алгоритмов, благодаря которым ПК может обрабатывать тексты. Для упрощения обработки текстов на естественном языке компьютером часто размечают текст специальным образом, использую те или иные стандарты или форматы верстки.

В качестве стандартов описания данных можно назвать XML – один из первых расширяемых языков разметки текста. Сам по себе XML не несет семантической нагрузки, а предназначен для упрощения машинной обработки текста при помощи специально созданных программных алгоритмов.

В базовой модели Семантической паутины, изначально предложенной Тимом Бернерсом-Ли, явно не было выделено наличие средств описания метаданных. Тем не менее в своих дальнейших работах, а также в работах других ученых указывается на важность включения в концепцию Семантической сети понятия метаданных. «Метаданные являются данными, предназначенные для идентификации, описания или локализации (местоположения) информационных ресурсов, независимо от физической природы ресурса».

Сам контент электронной библиотеки можно разделить на две части: сами материалы (материалы/документы) и их метаданные. Метаданные описывают свойства того или иного материала библиотеки, показывая разную информацию о нем – сведения о наименовании, авторе, издательстве и любые другие. Сами материалы – это непосредственно те ресурсы, которые нужны пользователям, то есть книги, статьи и любой другой контент [23]. Метаданные предназначены для упрощения поиска, каталогизации и разделения ресурсов друг от друга.

Одним из важных преимуществ электронных библиотек [38] является возможность интеграции материалов (данных) из разных источников, в том числе из других библиотек. Под интеграцией данных обычно понимают совместимость форматов хранения метаданных в единой электронной библиотеке.

Метаданные выполняют ряд функций:

– сопровождение материалов электронной библиотеки дополнительными сведениями, такими как наименование, сведения об авторе и т. д.;

– формирование списка ключевых слов для упрощения поиска материалов;

– распределение доступа к ресурсам электронной библиотеки;

– создание перечня сведений об информационном ресурсе, позволяющих интегрировать материалы из разных систем и электронных библиотек;

Сами метаданные также имеют разные способы классификации. Как правило, они отличаются друг от друга полнотой детализации в описании материалов. Для примера можно разделить метаданные по следующим типам:

– структурные;

– описательные;

– административные.

Структурные метаданные применяют для обеспечения навигации и поиска материалов в электронных библиотеках. В качестве структурных метаданных могут выступать сведения о каталогах или рубриках, к которым принадлежит тот

или иной материал, а также и описание частей самого материала внутри, например, разделение книги на главы внутри одного материала.

Описательные метаданные составляют основную информацию о мета атрибутах материалов цифровой библиотеки. Сведения о наименовании, авторе, издательстве и многие другие – все это описательные метаданные. Обычно поиск по библиотеке выполняют именно со вводом описательных метаданных.

Административные метаданные представляют собой информацию о правах на сам материал, адресе места хранения, описывают уровни доступа к материалу [54] и т. д.

Метаданные в библиографических информационных ресурсах представляют собой основные факты о каждом элементе (книге, монографии, публикации и т. д.) библиотеки.

При большом количестве источников возрастает сложность процесса интеграции за счет разницы в форматах хранения метаданных [50]. Форматы хранения определяются стандартами хранения, наиболее распространенными среди которых являются Дублинское ядро (Dublin Core) и MARC (Machine-Readable Cataloging, «машиночитаемая каталогизация»). Оба стандарта широко распространены в сфере описания полей информационных ресурсов, содержащих библиографическую или схожую с ней информацию.

Dublin Core представляет из себя набор элементов метаданных для представления разных информационных ресурсов, созданный для обеспечения глобального взаимодействия приложений, работающих с метаданными, – The Dublin Core Metadata for Simple Resource Discovery. Основные понятия остаются неизменными в рамках предметной области, что позволяет унифицировать большой диапазон информационных ресурсов. В качестве информационного ресурса в рамках стандарта понимается какой-либо идентифицируемый объект, который используется для хранения, обработки и передачи информации согласно рекомендации интернет RFC 2396 URI.

Начиная с 2005 года Dublin Core представлен в формате RDF (Resource Description Framework) [82], модели описания связанных данных, позволяющей структурированно описать любую сущность, благодаря чему RDF стал популярной основой для обозначения элементов в Семантической паутине.

В рамках Dublin Core выделяются 9 типов информационных объектов:

1. Коллекция. Представляет собой массив ресурсов, к каждому из которых может быть осуществлен отдельный доступ.

2. Данные. Внутри возможно хранение данных в разных определенных форматах (таблицы, гипертекст, БД и др.). Единство формата позволяет осуществлять программную обработку материала.

3. Событие. Могут содержать сведения о каких-либо мероприятиях или явлениях. Для описания можно использовать такие метаданные, как длительность, местоположение и др.

4. Изображение. Статичный мультимедийный объект. В качестве подобного объекта может выступать рисунок, фотография, географическая карта и т. д.

5. Интерактивный объект. Мультимедийный или иной объект, с которым можно взаимодействовать для просмотра и изменения его свойств. Например: видеофайл, интерактивная форма и т.п.

6. Сервис. Подсистема, выполняющая какую-либо или ряд функций. Веб-сайт также можно считать сервисом.

7. Программные средства. Скомпилированный или интерпретируемый программный код, пригодный для запуска на различных ЭВМ.

8. Аудио. Элемент, который содержит звуковую информацию.

9. Текст. Элемент для чтения его человеком. Этот элемент может быть статьей, книгой, монографией и так далее. Оцифрованная копия страницы (скан-копия) документа также считается текстом.

Dublin Core состоит из 15 базовых элементов метаданных, которые можно дополнять различными квалификаторами.

Базовыми элементами Dublin Core являются:

– Title – название;

– Creator – создатель;

– Subject – тема;

– Description – описание;

– Publisher – издатель;

– Contributor – внесший вклад;

– Date – дата;

– Type – тип;

– Format – формат документа;

– Identifier – идентификатор;

– Source – источник;

– Language – язык;

– Relation – отношения;

– Coverage – покрытие;

– Rights – авторские права.

Элементы могут быть использованы в разном порядке, а также повторяться, что позволяет детально описать ресурс. Компактность, доступность и актуальность стандарта обеспечили его широкое распространение.

MARC является стандартом для представления и передачи библиографической информации в машиночитаемой форме. В отличие от упомянутого выше стандарта, используемого для описания электронных ресурсов, MARC используется для описания печатных изданий. На сегодня существует несколько форматов в рамках MARC, основными из которых являются форматы UNIMARC и MARC21. Изначально формат MARC был разработан в 1960-е годы сотрудниками Библиотеки Конгресса США [4] для автоматизации преобразования каталожных карточек с метаданными книг [10]. Позднее были созданы варианты формата MARC для других видов материалов: периодики, карт, нот и т. д., а также разные версии, которые имели ориентацию на национальные правила каталогизации для материалов разных стран. Развитие MARC привело к появлению

как национальных, так и международного коммуникативного формата. Возникновение и дальнейшее развитие форматов MARC дало возможность упростить процесс обработки изданий и сосредоточить его в центрах компьютерной каталогизации.

Распространение стандарта MARC сделало возможным создание сводных электронных каталогов, в которых сейчас хранятся десятки и сотни миллионов записей. MARC является удобным инструментом для разметки любой части машиночитаемой каталогизационной записи.

Международный формат MARC (UNIMARC) был создан для того, чтобы обойти несовместимость форматов [60]. UNIMARC позволил принимать записи, созданные в любом формате MARC, преобразовывать их в UNIMARC, а из него – в любой другой национальный формат MARC. Из этого следует, что основной целью создания UNIMARC является помощь международному обмену библиотечным данными в машиночитаемой форме между разными национальными библиографическими службами [47].

UNIMARC предназначен для описания текстовых документов (таких как книги), периодики, электронных ресурсов и мультимедийной информации, например изображений.

Поля UNIMARC бывают двух типов: специфические и общие. В отличие от общих, используемых для универсального описания разных типов материалов, специфические применяют для конкретно обозначенных типов документов.

Существует и отдельный подвид UNIMARC, включающий в себя правки для соответствия современным нормативным правилам учета электронных документов. Данный формат называется RUSMARC [43].

RUSMARC, как и UNIMARC, не оговаривает тип или внутренний контент материалов, но включает конкретные рекомендации по описанию метаданных для их последующей обработки во внешних системах.

Для удобства использования Библиотека Конгресса США создала также структуру xml-схемы, основанную на MARC21, – MARC XML. Структура является

гибкой и расширяемой для того, чтобы пользователи могли работать с данными формата MARC способами, соответствующими их потребностям. Структура содержит многие компоненты, такие как схемы, таблицы стилей и программные средства, разрабатываемые и поддерживаемые Библиотекой Конгресса.

MARC XML может использоваться следующим образом:

– для представления полной записи MARC в XML;

– как схема расширения для METS (стандарт кодирования метаданных и передачи);

– для представления метаданных для инициативы «Открытые архивы» (организация, разрабатывающая и применяющая стандарты технической совместимости для архивов для обмена информацией каталога (метаданные));

– для исходного описания ресурса в синтаксисе XML.

Схема MARC XML поддерживает все кодированные данные MARC независимо от формата. Более того, MARC XML – это компонентная, расширяемая архитектура, позволяющая пользователям подключать и воспроизводить разные части программного обеспечения для создания пользовательских решений. Стоит однако отметить, что проверка MARC не выполняется схемой, а должна быть осуществлена внешним программным обеспечением.

Ядром XML-схемы MARC является простая XML-схема, содержащая данные MARC. Эта базовая схема может использоваться, когда необходимы полные записи MARC или как шаблон для дальнейших преобразований, например, в Dublin Core или для других процессов, например, валидации (проверки). XML-схему MARC не нужно редактировать, чтобы отразить незначительные изменения в MARC21. Схема сохраняет семантику MARC.

Оба стандарта достаточно распространены, что приводит к проблеме интеграции данных из разных электронных библиотек, так как разные информационные ресурсы могут использовать разные стандарты и форматы хранения. При этом если для форматов MARC существуют конвертеры данных,

позволяющих быстро перевести данные из одного формата в другой за счет их схожести, то конвертация из MARC в Dublin Core является более трудоемкой. Кроме того, узкоспециализированные и предметно-ориентированные библиотеки зачастую вносят изменения в стандарты, создавая собственные форматы, так как унифицированные форматы являются недостаточными для специфических нужд подобных библиотек.

Таким образом, при создании электронной библиотеки необходимо определить параметры проектирования электронной библиотеки, а также выбрать схему представления метаданных и учесть ее в архитектуре электронной библиотеки.

Выводы по главе: автор привел базовые определения основных терминов и описал краткую историю появления и развития электронных библиотек. Были рассмотрены методы хранения данных в электронных библиотеках, описаны основные характеристики, процессы и различия электронных библиотек от традиционных. Приведены изначальные принципы при проектировании автоматизированной библиотечно-информационной среды. Даны определения метаданных, стандартов и форматов их хранения, а также типизация информационных объектов.

ГЛАВА 2. ОБОСНОВАНИЕ ПРИНЦИПОВ КОНСТРУКТОРА ПОЛЕЙ ИНТЕГРАЦИИ ИНФОРМАЦИИ И МОДЕЛИ ИЗВЛЕЧЕНИЯ МЕТАДАННЫХ ИЗ ПОЛНОТЕКСТОВЫХ ДОКУМЕНТОВ

Развитие электронных библиотек позволяет обеспечить доступ широких слоев населения к достоверной информации [11], однако, как и к любым информационным системам в современном обществе, к ним предъявляются жесткие требования со стороны пользователей. В условиях доступности информации пользователям необходимо получать нужные сведения с максимальной скоростью, при этом информация должна точно соответствовать пользовательскому запросу [22].

Для соответствия электронной библиотеки предъявляемым требованиям необходимо использование современных технологий и продуманного подхода к проектированию базы данных электронной библиотеки. В данной главе рассмотрена концептуальная схема электронной библиотеки FRBR, описаны варианты связей в базе данных электронной библиотеки, предложена структурная схема электронной библиотеки, позволяющая обеспечить соответствие библиотеки предъявляемым требования, а также рассмотрена проблема интеграции данных из разных источников и предложена модель конструктора правил интеграции информации из распределенных разнородных источников [65], позволяющая упростить процесс наполнения базы данных электронных документов.

2.1 Предлагаемая структура электронной библиотеки

Электронные библиотеки объединяют разные виды данных, при этом данные должны быть структурированы и систематизированы [61]. Электронная библиотека должна обеспечивать наиболее полный набор доступных ресурсов,

обеспечивая универсальность данных, при этом навигация и поиск внутри библиотеки должны отвечать следующим требованиям [62]:

– минимальное время отклика на запрос;

– интуитивно понятный интерфейс;

– удобство использования;

– возможность индексирования страниц библиотеки в рамках Семантической паутины.

Для описания структуры баз данных обычно используют ER-модели (entity-relationship model) и метод ER-диаграмм [69]. Эти модели описывают концептуальные схемы базы данных. IFLA разработала ER-модель для описания базы данных электронной библиотеки [56]. Эта схема имеет название FRBR [94]. Внутри нее содержится описание всех сущностей (таблиц базы данных), их свойства и связи сущностей друг с другом, которые могут потребоваться для создания собственной электронной библиотеки.

В описанной модели имеется 3 типа таблиц:

1) таблицы для объектов,
2) таблицы для субъектов,
3) таблицы для описателей объектов.

В FRBR подробно описаны все сущности, но сведения о материалах библиотеки содержатся в таблицах для объектов, поэтому им уделено особое внимание. В них содержатся сведения о самом материале и его метаданных.

Связи сущностей разных типов показывают их отношение друг к другу. Например, показана связь между конкретно описанным материалом и конечным файлом, хранящимся на диске.

ER-модель FRBR показывает максимально общие правила для описания библиографической информации и на верхнем уровне определенно может подойти для описания электронной библиотеки. Однако абсолютно универсальным решением FRBR назвать нельзя, так как она не учитывает никакой конкретной

специфики, которая может быть необходима в предметно-ориентированных электронных библиотеках.

Стоит отметить, что модель FRBR разрабатывалась с учетом конкретных пользовательских задач:

– поиск – нахождение одной сущности или набора сущностей в результате поиска, используя атрибуты или отношения сущностей;

– идентификация – подтверждение, что описанная сущность точно отвечает искомой сущности, или что существуют различия между двумя или более сущностями со схожими характеристиками;

– выбор – выбор сущности, которая наиболее полно отвечает требованиям пользователя с учетом содержания, физического формата и т. д.; отказ от сущности, которая не соответствует потребностям пользователя;

– получение – получение сущности через покупку, заем и т. д. или получение доступа к сущности в электронном виде онлайн.

Модель FBRB является концептуальной и позволяет описать библиотечный фонд на уровне планирования. С помощью этой модели могут быть определены основные сущности, атрибуты и отношения между объектами, однако для построения электронной библиотеки необходимо определение более полной модели базы данных.

Определение основных параметров электронной библиотеки необходимо для обеспечения оптимальной структуры базы данных электронной библиотеки [51]. Для использования всех возможностей электронной библиотеки предлагается использование универсальной связанной базы данных, включающей в себя возможность подключения внешних рубрикаторов и работу с разными форматами и словарями хранения данных [77]. Автором была разработана ER-схема электронной библиотеки, позволяющая удовлетворить все обозначенные потребности и совместимая со стандартом RUSMARC. Основными элементами предлагаемой схемы электронной библиотеки являются материалы, источники и авторы. Материалы могут быть объединены в разные категории, а также

распределены по коллекциям. В случае создания узкоспециализированных библиотек, таких как научные или педагогические библиотеки, материалы и авторы связываются с организациями, в рамках которых был разработан тот или иной материал.

Общая схема основных таблиц базы данных представлена на

Рисунок 1.

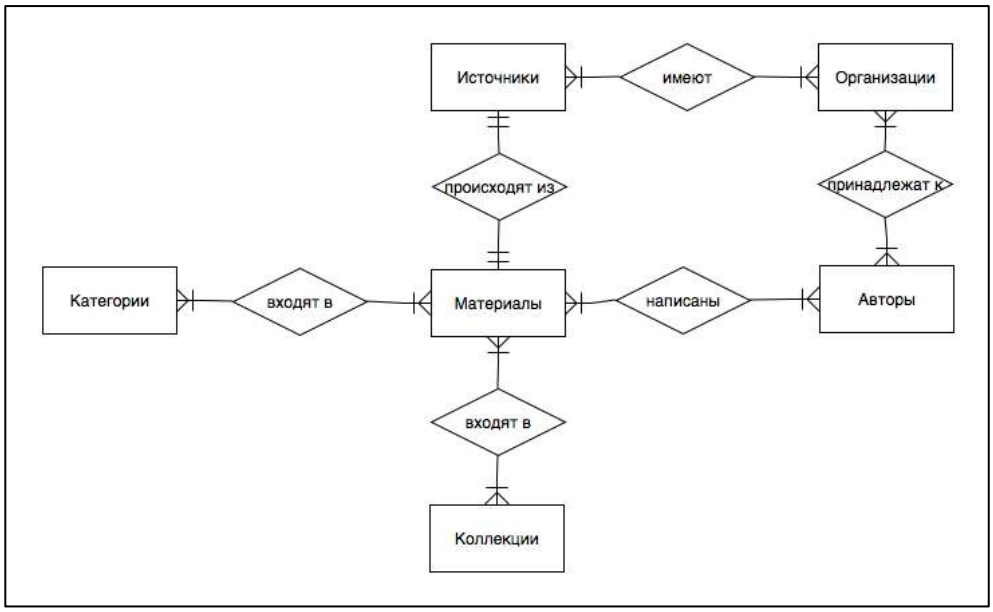


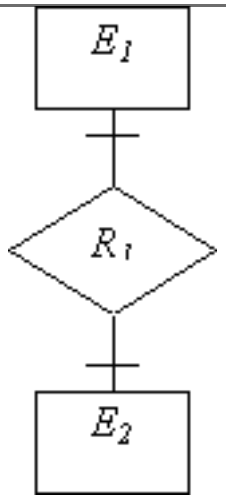
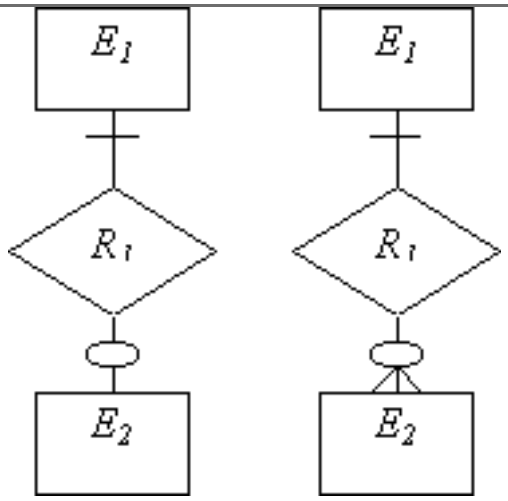
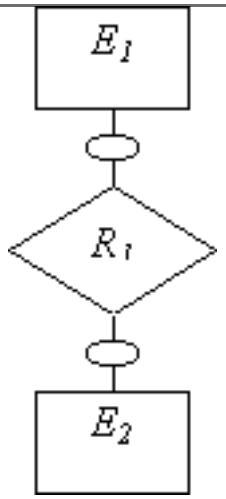
Рисунок 1. Предлагаемая структурная схема электронной библиотеки

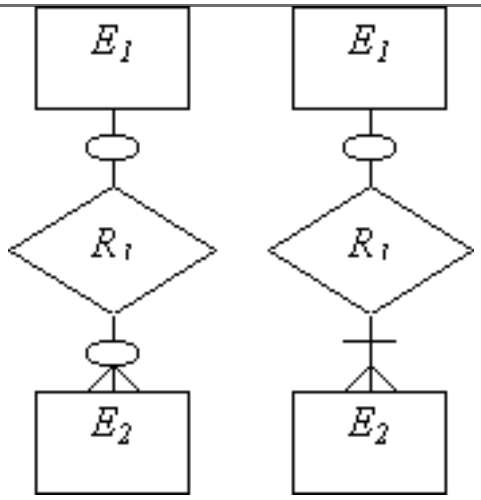
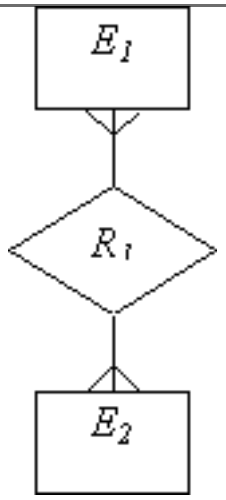
Как видно из рисунка 1, в предлагаемой структуре используются те же элементы, что и в модели FRBR, но структура более детализирована и содержит концепцию объединения отдельных элементов в упорядоченную базу данных.

Предлагаемая схема позволит решить большинство из указанных ранее проблем. Описание возможностей указанной схемы представлено в этой главе.

Навигация в электронной библиотеке осуществляется за счет связей между таблицами базы данных. В таблице 1 представлены возможные типы связи, использующиеся при построении базы данных.

Таблица 1. Типы связей базы данных

Тип связи	Пример связи	Правило построения отношений
(1,1):(1,1)		Требуется только одно отношение. Первичным ключом данного отношения может быть ключ любой из сущностей.
(1,1):(0,1) (1,1):(0,n)		Для каждой сущности создается свое отношение, при этом ключи сущностей служат ключами соответствующих отношений. Кроме того, ключ сущности с обязательным классом принадлежности добавляется в качестве внешнего ключа в отношение, созданное для сущности с необязательным классом принадлежности.
(0,1):(0,1)		Необходимо использовать три отношения: по одному для каждой сущности (ключи сущностей служат первичными ключами отношений) и одно отношение для связи. Отношение, выделенное для связи, имеет два атрибута – внешних ключа – по одному от каждой сущности.

Тип связи	Пример связи	Правило построения отношений
(0,1):(0,n) (0,1):(1,n)		Формируются три отношения: по одному для каждой сущности, причем ключ каждой сущности служит первичным ключом соответствующего отношения, и одно отношение для связи. Отношение, выделенное для связи, имеет два атрибута – внешних ключа – по одному от каждой сущности.
n : m		В этом случае всегда используются три отношения: по одному для каждой сущности, причем ключ каждой сущности служит первичным ключом соответствующего отношения, и одно отношение для связи. Последнее отношение должно иметь среди своих атрибутов внешние ключи, по одному от каждой сущности.

При типе связи «многие ко многим» достигается наиболее полная связь таблиц, таким образом, возможно обеспечить гибкую систему доступа пользователя к данным, включая возможность разного распределения данных.

В предложенной структуре базы данных электронной библиотеки между всеми таблицами существует связь «многие ко многим» за счет использования промежуточных таблиц. Такая структура позволяет обеспечивать максимальную навигацию в рамках библиотеки, извлекать любые данные из таблиц, а также формировать сложные запросы.

Для обеспечения максимального удобства работы с электронной библиотекой помимо основных таблиц базы данных возможно хранение данных о зарегистрированных пользователях, включая связь с материалами, которые

пользователь добавил в избранное, доступ к комментированию материалов, публикацию ссылок на материалы в сторонних ресурсах. Эти функции позволяют сделать электронную библиотеку максимально ориентированной на пользователя, что особенно важно для распространения достоверной информации [64].

Предложенная структура также подразумевает наполнение библиотеки как собственными ресурсами, так и интеграцию с другими базами данных, что позволяет обеспечить максимальную наполняемость библиотеки и, как следствие, ее универсальность и разнообразность. База данных содержит все поля формата MARC, что обеспечивает ее совместимость с большинством внешних ресурсов, так как этот формат является одним из самых используемых в электронных библиотеках [65]. Для обеспечения совместимости с другими форматами возможно добавление кастомизированных полей, для чего в структуре базы существует отдельная таблица. Наличие полей формата MARC обеспечивает хранение метаданных в виде, пригодном для машинного прочтения, что позволяет индексировать страницы библиотеки, а также внешним агрегаторам использовать технологии Семантической паутины.

Предлагается использование открытого доступа [28, 29, 83, 86] с возможностью регистрации для доступа к расширенному функционалу (например, добавление материалов в избранное).

Такая структура базы данных предполагает обеспечение следующего функционала:

– хранение электронных версий бумажных материалов, включая атрибутивную информацию (метаданные) о предоставляемых материалах и их авторах;
– хранение базы авторов работ;
– хранение видеоматериалов, включая атрибутивную информацию (метаданные) о предоставляемых материалах и их авторах;
– добавление новых записей и работа с уже существующими записями (материалы, авторы, организации, медиафайлы);
– хранение базы организаций;

– ведение статистики материалов, авторов, источников, организаций;

– распределение материалов в соответствии с их классификацией для обеспечения удобства навигации;

– распределение материалов по коллекциям;

– разделение материалов по виду;

– поиск по записям в библиотеке;

– регистрация и вход в личный кабинет пользователя;

– хранения избранных записей зарегистрированных пользователей на сайте;

– обмен мнениями о материалах библиотеки (комментарии);

– публикация ссылок на материалы в социальных сетях.

На

Рисунок 2 представлена часть структуры предлагаемой электронной библиотеки, относящаяся к источникам данных.

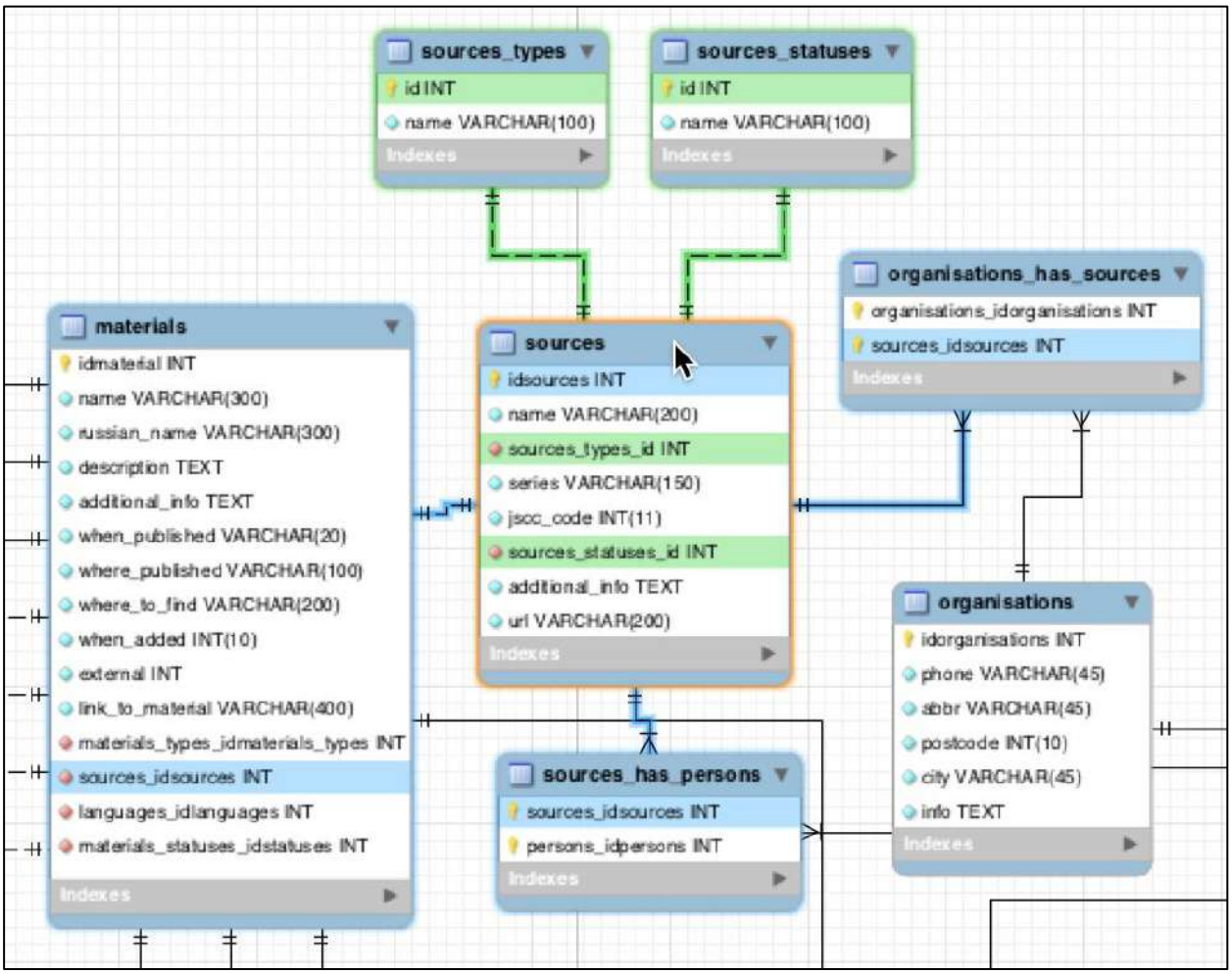


Рисунок 2. Источники материалов

Как видно из предложенной схемы, источников материалов, ключевым полем для таблицы источников является его уникальный идентификатор. Для каждого источника может быть определено название, серия, код, ссылка, дополнительная информация. В отдельных таблицах хранятся тип и статус источника. Каждый источник имеет связь с организацией, материалом и автором. Под источником понимается издательство, выпустившее материал или аналог издательства (журналы, газеты и др. [89]). Определение источника материала позволяет быстро найти материалы специализированных изданий, что является значительным преимуществом в случае, если электронная библиотека является предметно ориентированной.

На рисунке 3 представлена часть структуры базы данных, относящаяся к авторам. Ключевым полем таблицы является уникальный идентификационный номер, который используется для соединения с другими таблицами со связями «многие ко многим», что позволяет определить, какие материалы публиковал автор, в каких источниках, в каких организациях автор работал или работает на данный момент.

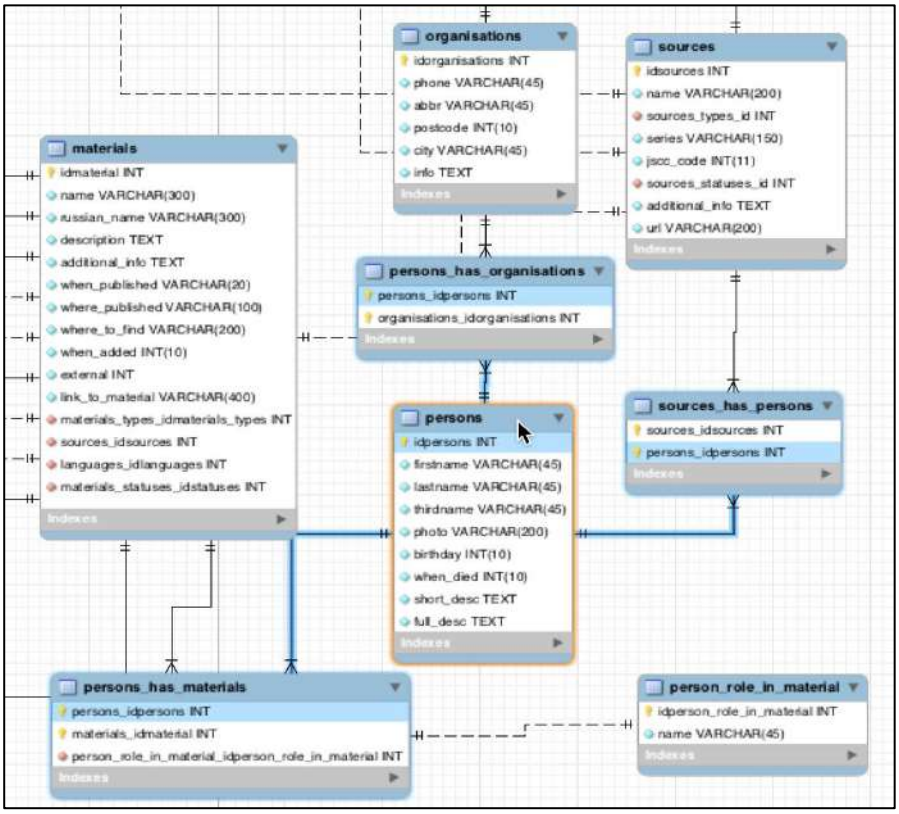


Рисунок 3. Информация об авторах

Информация об авторе включает поля для имени, фамилии и отчества автора, фотографию автора, дату рождения и дату смерти, краткое и полное описание для размещения на странице автора. Поиск возможен по всем указанным полям. Возможно создание сложных поисков, включающих информацию как из полей таблицы автора, так и из связанных таблиц.

На Рисунок 4 представлена часть структуры базы данных, относящаяся к материалам. Ключевым полем таблицы является уникальный идентификационный номер. Таблица содержит поля с именем автора, русскоязычным написанием имени автора, описанием материала, дополнительной информацией, данными о публикации (где и когда было опубликовано), где материал можно найти, когда материал был добавлен в библиотеку, ссылку на материал.

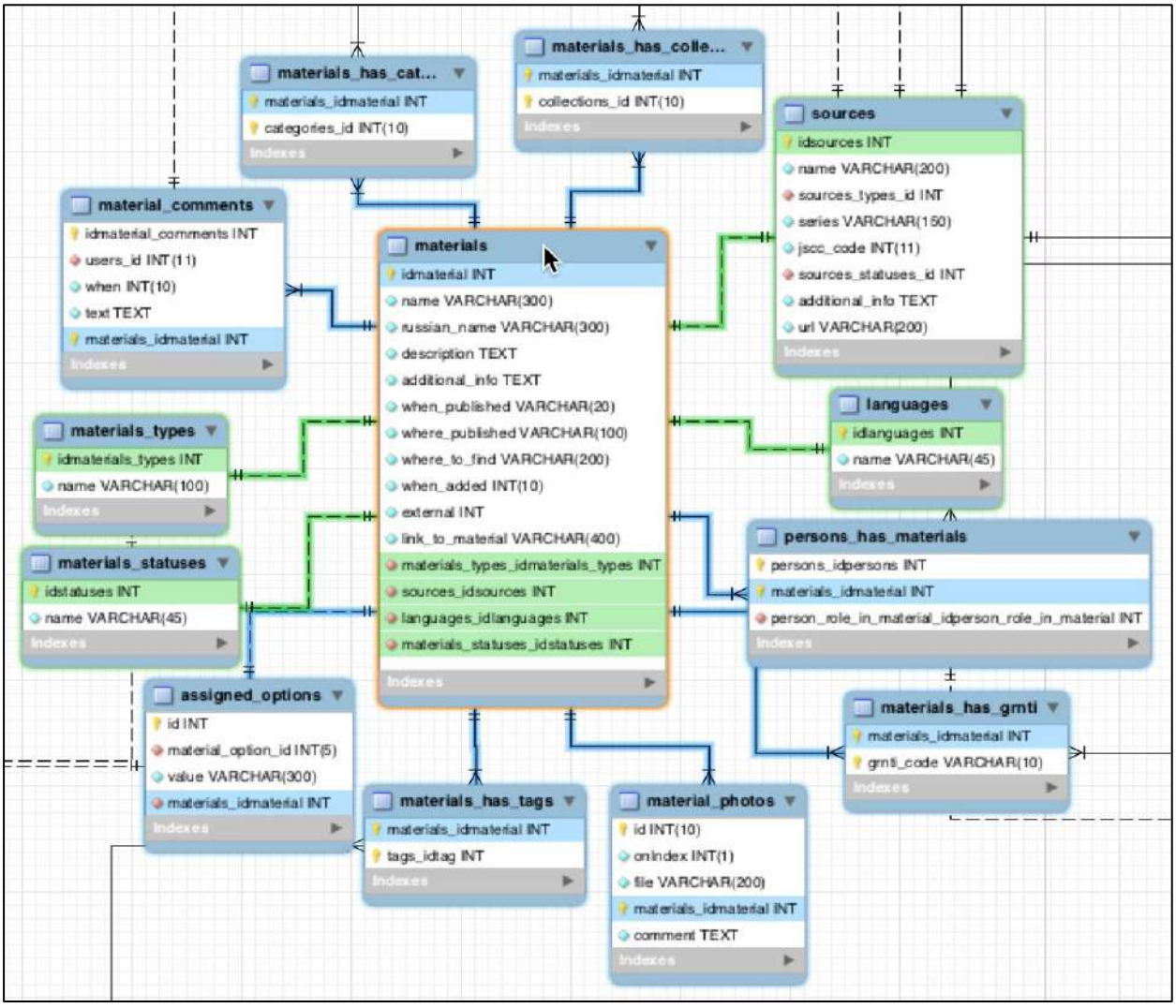


Рисунок 4. Материалы библиотеки

Помимо информации, содержащейся непосредственно в таблице материалов, дополнительные таблицы позволяют определить статус материала, тип, опции, тэги и фотографии материала.

Как видно из элементов структуры базы данных, связи «многие ко многим» позволяют наиболее полно извлекать информацию из электронной библиотеки, например, возможно извлечение следующей информации:

– все материалы, написанные внутри одной организации (через связь: материал – персона – работает в организации);

– рубрики ГРНТИ, по которым публиковался автор (через связь: персона – материалы – ГРНТИ);

– материалы про конкретного человека (а не написанные им, т.е. биографии о нем) (через связь: персоны – материалы – роль персоны в материале).

Таким образом, с помощью предложенной структуры возможно максимальное удовлетворение потребностей посетителей электронной библиотеки за счет возможности получения информации с высокой эффективностью. Возможность привлечения внешних ресурсов для наполнения библиотеки позволяет обеспечить наиболее полный доступ читателей к информации [66].

Предлагаемая структура электронной библиотеки может быть использована как для сверхуниверсальных библиотек, так и для узкоспециализированных. Возможность настройки таблиц, добавления и удаления полей, а также настройки доступа пользователей к библиотеке позволяет осуществить любые вариации доступа к ресурсам, а также обеспечить наиболее удобную для пользователя навигацию. Так, при создании любого типа библиотеки возможность объединения материалов в категории и коллекции позволяет упростить процесс поиска материалов в библиотеке, а возможность поиска с учетом связей между таблицами гарантирует нахождение пользователем искомого материала. В результате

возрастает вероятность использования материалов, находящихся в библиотеке, в сравнении с библиотеками, где не реализована предлагаемая схема, так как быстрое нахождение материала является критичным для использования ресурса в век развития интернета и возрастающих объемов информации.

Универсальность электронной библиотеки может быть достигнута с помощью интеграции собственных ресурсов со сторонними ресурсами [63]. Проблема интеграции будет подробнее рассмотрена в разделе 2.2, однако стоит отметить, что интеграция ресурсов с помощью существующих способов не всегда доступна. Предлагается дополнить автоматизированную библиотечно-информационную среду конструктором правил интеграции данных из различных распределенных источников (базы данных, веб-сайты и документы в формате PDF без прилагаемых метаданных), разработанным в рамках данной работы и описанным в разделе 2.3.

Согласно многочисленным исследованиям, пользователи чаще всего закрывают сайт и обращаются к другим источникам при больших промежутках времени ответа на запрос [55]. Развитие информационных технологий позволяет получать информацию все быстрее, и при доступности большого количества источников возникает конкуренция между ними [76].

В стандартной теории различают три лимита времени ожидания:

1. 0,1 секунды – при таком времени ожидания у пользователя создается ощущение, что процессом управляет сам пользователь, а не компьютер. Это идеальное время ответа системы на действия пользователя.
2. 1 секунда – лимит времени ожидания, при котором пользователь ощущает, что он управляет процессом, при этом не ожидая действий компьютера слишком долго. Задержка в 0,2–1 секунду означает, что пользователи замечают, что компьютер выполняет действие.
3. 10 секунд – лимит времени ожидания, при котором пользователь остается сконцентрированным на задаче. При времени ожидания

более 10 секунд требуется наличие указания оставшегося времени выполнения операции (или указания того, на сколько процентов выполнена задача) и кнопки прерывания операции.

Несмотря на то, что эти лимиты справедливы для человеческой психики, при возможности совершения той же операции на другом сайте или в другой программе пользователь скорее всего выберет более быстрый вариант [24]. Среди разработчиков веб-сайтов принято считать, что нормальное время ответа на запрос пользователя должно составлять не более секунды.

Предложенная схема организации автоматизированной библиотечно-информационной среды позволяет обеспечить наиболее быстрое время ответа на пользовательский запрос. Для проверки этого утверждения был проведен эксперимент по проведению запроса к базе данных при разных типах связи. Были получены следующие результаты:

– связи «один ко многим» – 0,49 мс;

– связи «многие ко многим» – 0,26 мс;

– связи «один к одному» – 0,74 мс.

2.2 Интеграция данных в электронных библиотеках

Существует ряд способов интеграции данных [53, 66] в тех случаях, когда форматы хранения совпадают или же достаточно описаны для создания специализированного конвертера, таких как:

1) интеграция на уровне полей базы данных;
2) использование API;
3) получение метаданных по коду ISBN;
4) синтаксический разбор HTML страниц исходной электронной библиотеки;
5) извлечение метаданных непосредственно из текстов материалов исходной электронной библиотеки.

Интеграция на уровне полей базы данных подразумевает под собой описание соотношения полей исходной и целевой баз данных, для чего необходимо иметь доступ к исходной базе данных для детального описания ее структуры. Однако в случае, если структура исходной базы данных неизвестна, этот способ неосуществим [77].

API (Application programming interface – интерфейс программирования приложений) представляет собой набор готовых классов, процедур, методов, функций и констант, предоставляемых сторонним приложением (библиотекой, порталом) для использования их в других программных системах. При использовании такого интерфейса целевая библиотека должна получать из исходной библиотеки данные в требуемом формате. Наиболее распространенными форматами являются:

– XML – расширяемый язык разметки. XML является рекомендованным Консорциумом Всемирной паутины (W3C) языком разметки. Спецификация XML описывает документы и частично описывает поведение XML-обработчиков (утилит, позволяющих читать XML-документы и осуществляющих доступ к их содержимому). XML создавался как язык с простым формальным синтаксисом, подходящим для создания и чтения документов компьютерными программами и одновременно удобный для восприятия и создания документов человеком. При этом основным направлением для использования XML является интернет. XML называется расширяемым из-за того, что он не ограничивает разметку, используемую в документах: разработчик может создать разметку сообразно с потребностями к выбранной области, являясь ограниченным только синтаксическими нормами языка. Комбинирование простого и понятного формального синтаксиса, удобства для разработчика, расширяемость, а также базирование на международных кодировках типа Юникод для хранения содержания документов способствовало широкому использованию как базового XML,

так и множества специализированных языков, основанных XML в различных программных средствах.

– JSON (JavaScript Object Notation) – текстовый формат, используемый для обмена данными между приложениями. Основан на программном языке JavaScript. Как и множество других текстовых форматов, JSON довольно просто воспринимается человеком. Невзирая на происхождение от языка JavaScript (стандарта ECMA-262 1999 года), формат является языконезависимым и может быть использован с любыми языками программирования. Для разных языков существуют готовые библиотеки для составления и обработки данных в формате JSON.

– Plain Text (обычный текст без разметки) – самый сложный для интеграции вариант, так как за счет отсутствия разметки невозможно однозначно интерпретировать и интегрировать поля. В таком случае необходимо осуществлять семантический разбор текста, например, при помощи GLP-парсера, позволяющего написать правила обработки полного текста для разделения его на элементы.

Использование API является удобным способом интеграции данных одной библиотеки в другую, однако для этого исходная библиотека должна предоставить такую возможность [57, 87].

ISBN (International Standard Book Number – международный стандартный книжный номер) – уникальный код книжного издания, разработанный в Великобритании на базе 9-значного стандартного книжного номера (Standard Book Numbering code). В 1970 году ISBN с небольшими изменениями был принят как международный стандарт ISO 2108. 1 января 2007 года создан новый стандарт ISBN – 13-значный, идентичный штрих-коду. При использовании ISBN для передачи данных исходная библиотека должна передать код в целевую, после чего возможно получить метаданные материалов благодаря использованию поиска по кодам, например, через Google Books ISBN API (Рисунок 5).

Рисунок 5. Пример использования Google Books ISBN API для получения метаданных по коду ISBN

Синтаксический разбор HTML-страниц исходной электронной библиотеки является одним из самых трудоемких вследствие необходимости создания парсера (синтаксического анализатора) HTML-кода [45] для каждой исходной библиотеки, а также робота для автоматизированного обхода страниц целевой библиотеки. Для упрощения задачи и избавления от необходимости написания робота можно использовать API поисковых систем (Yandex, Google) и формировать перечень страниц для синтаксического анализа из их индекса, однако этот способ возможен только для открытых и общедоступных электронных библиотек [5], у которых разрешена индексация поисковыми системами.

Извлечение метаданных непосредственно из текстов материалов исходной электронной библиотеки является наиболее трудоемким способом, однако единственно возможным в случае, если исходная электронная библиотека является простейшим файловым хранилищем и не использует какой-либо библиотечный стандарт хранения данных. Обязательным условием является наличие внутри

материалов полных текстов (HTML, текстовая подложка внутри PDF и пр.) либо возможность извлечения текста благодаря оптическому распознаванию (OCR – optical character recognition). После извлечения текста из материала необходимо провести его семантический разбор и сформировать итоговые метаданные, которые будут загружены в целевую библиотеку [85].

Возможна и обратная ситуация – целевая библиотека может создать API или механизм передачи/наполнения данных на своей площадке (в качестве примера можно привести европейскую цифровую библиотеку Europeana [86, 91] или российский проект «Научное наследие России» [27, 59, 88, 93]), и принимать материалы и их метаданные от исходных библиотек. В таком случае исходные библиотеки со своей стороны должны подготовить данные в нужном виде. Способы обмена данными в таком случае соответствуют способам обмена (форматам), описанным ранее.

Несмотря на попытки унификации библиографических метаданных, наблюдается ряд проблем при интеграции материалов разных библиотек, возникающих в основном из-за разных форматов хранения метаданных [74]. Существуют способы интеграции метаданных между электронными библиотеками, однако в большинстве случаев интеграция невозможна без предварительной работы над совместимостью форматов и/или полей библиотек, при этом если исходная библиотека не обеспечивает предоставление метаданных в удобной для интеграции форме, процесс интеграции может занимать значительное время. При необходимости одновременной интеграции из нескольких источников длительность интеграции может увеличиваться в разы.

Таким образом, предложенная схема структуры электронной библиотеки обеспечивает эффективную навигацию, минимальное время ответа на запрос пользователя и возможность интеграции с материалами других библиотек способами, описанными выше. Однако при необходимости интеграции материалов более чем из одного источника процесс является трудоемким. Для решения этой задачи автором был создан и внедрен конструктор полей интеграции данных.

2.3 Решение задачи интеграции данных из разных источников

В качестве первого этапа создания унифицированного конструктора, позволяющего задавать правила для интеграции мета-атрибутов из внешних материалов, потребуется провести анализ форматов хранения материалов и полей во внешней библиотеке. Если имеется возможность доступа напрямую ко внешней базе данных (используя язык SQL – Structured Query Language), то станет возможным делать любые выборки данных из внешней базы и представлять метаданные в требуемом виде. Однако чаще всего доступ напрямую к базе данных отсутствует, зато внешние порталы могут иметь специальные средства для обмена данными – API (application programming interface). API представляет собой механизм, реализующий средства обмена данными по каким-либо объединенным правилам, используя протоколы HTTP и HTTPS. Многие крупные электронные библиотеки (например, Europeana) и такие проекты, как Google Books имеют подготовленные API со всей необходимой документацией. Внешним разработчикам остается только реализовать механизмы работы с этими API, что существенно упрощает процесс интеграции данных с этими системами.

Задача интеграции данных из внешнего источника становится существенно сложнее в том случае, если нет прямого доступа к SQL-серверу [80] и API целевой библиотеки. В случае если внешняя электронная библиотека доступна через HTTP/HTTPS, можно проводить синтаксический разбор исходной HTML-разметки библиотеки и повторять его для каждого уникального материала.

Еще одним вариантом, возможным для интеграции внешних данных при отсутствии всех выше обозначенных механизмов обмена информацией, может быть извлечение мета-атрибутов из полных текстов материалов, хранящихся на диске. Этот процесс описан в разделе 2.4 данной работы.

После успешного извлечения атрибутивной информации из исходных полей внешней электронной библиотеки будут получены базовые данные, требующие конвертации в целевые поля собственной базы данных. Для реализации механизма

конвертации будет необходима разработка модели, описывающей связи между исходными и целевыми полями наполняемой библиотеки.

В результате диссертационного исследования была спроектирована блок-схема механизма автоматизированного конструктора правил интеграции полей исходной и целевой базы данных электронной библиотеки (Рисунок 6). Блок-схема показывает возможные типы источника, необходимые для работы конструктора, в зависимости от наличия того или иного механизма для обмена данными в целевой библиотеке.

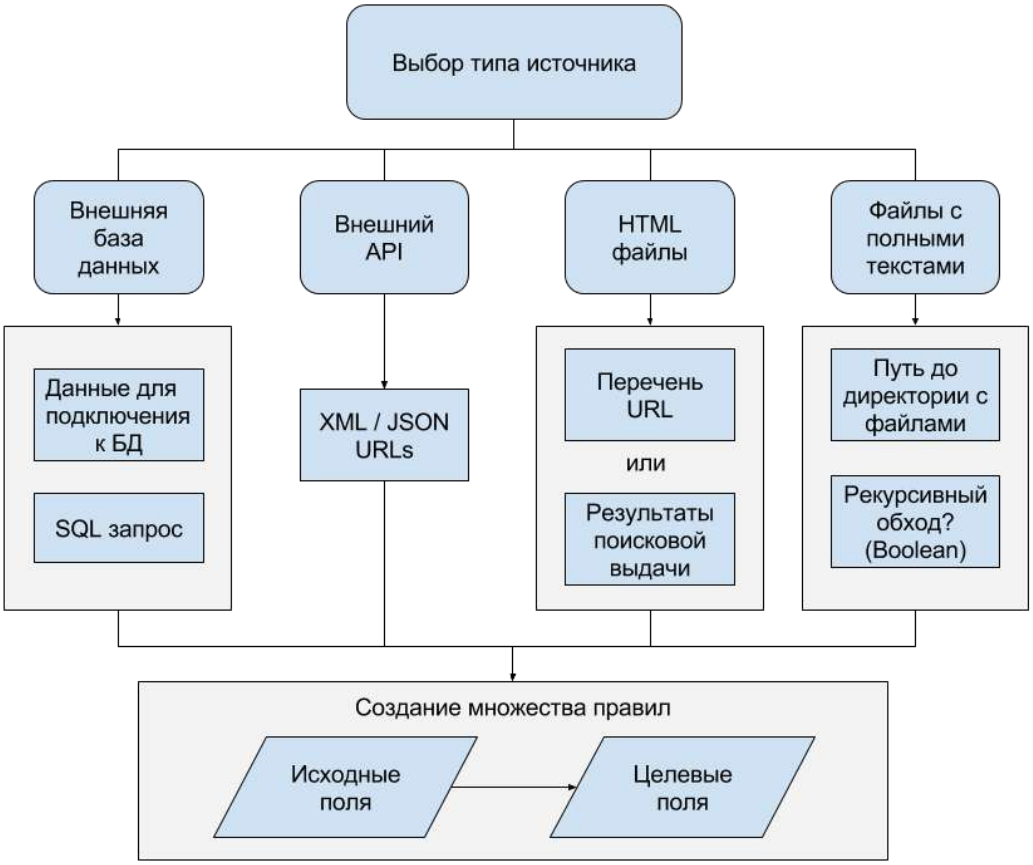


Рисунок 6. Возможные варианты интеграции в зависимости от типа источника

Сам по себе конструктор правил интеграции полей предлагается реализовывать в виде online-сервиса с пошаговым созданием правил для разных библиотек, которые требуется интегрировать.

Первым шагом при работе с конструктором является анализ исходной библиотеки для выявления требуемого типа данных. После определения типа обрабатываемых данных потребуется осуществить начальную конфигурацию конструктора, выбрав тип обрабатываемых данных (SQL / JSON / XML / HTML). При выборе HTML будет необходимо разобрать DOM (Document Object Model – «объектная модель документа») [96] – модель исходных страниц внешней библиотеки.

Важно отметить, что, в отличие от интеграции данных при помощи SQL или API, разбор DOM потребует детального изучения исходных кодов HTML-страниц внешней библиотеки. Так как в отличие от того же XML структура HTML дает намного большую свободу действий разработчику внешних электронных библиотек, их верстка может быть абсолютно разной от сервиса к сервису.

Проблемой также является то, что при изменении верстки страницы внешней библиотеки, пусть даже стилистической замены каких-то внешних элементов, может потребоваться повторная конфигурация для конструктора правил интеграции полей.

При разборе DOM модели HTML-документа вся страница может быть представлена в виде дерева тегов. Каждый тег может содержать внутри другие теги, текстовую, мультимедийную либо другую информацию. По дереву узлов можно перемещаться от родительских элементов к дочерним и обратно, обходя таким образом всю модель.

Фильтрация отдельных элементов внутри модели возможна по CSS (каскадные таблицы стилей)? селекторам (id, class, rel и другие) либо по уровню вложенности или относительно других элементов. Все эти механизмы лежат в основе технологии CSS.

После первоначального конфигурирования конструктора правил интеграции полей и назначения типа и формата исходных данных потребуется указание ссылки на сам источник исходных данных. В случае если у внешнего сервиса имеется API, становится возможным указать URL-идентификатор этого API. Для таких форматов, как JSON и XML потребуется указание одного или нескольких конкретных URL данных файлов. Важно отметить, что имеется возможность разбиения информации из источника постранично. На рисунке 7 продемонстрирован процесс выбора источника.

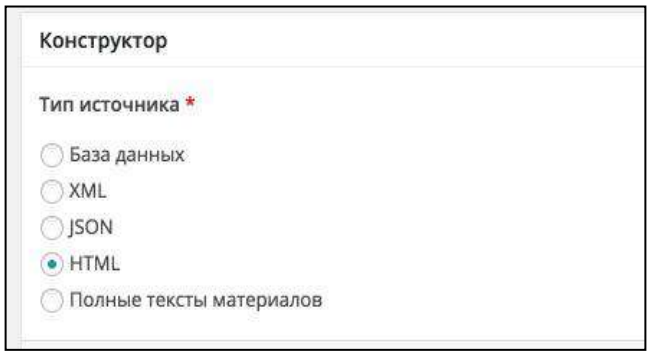


Рисунок 7. Выбор источника

После конфигурирования конструктора он формирует собственный API для передачи и дальнейшей работы с извлеченными метаданными внешних ресурсов. Доступ к этому API можно получить через HTTP-запрос типа POST.

Разработанный конструктор позволяет извлекать мета-атрибуты из исходных библиотек, хранящих и отображающих данные в одном из вышеперечисленных форматов. Тем не менее иногда возникает необходимость интеграции материалов, не сопровождаемых внешними метаданными из обычного файлового архива. Например, внешняя библиотека может прислать архив своих материалов без какого-либо сопроводительного описания.

Для решения задачи интеграции данных из файловой системы автором была исследована проблема извлечения метаданных из полнотекстовых оцифрованных материалов на примере документов формата Adobe PDF [64].

Алгоритм работы конструктора предполагает первоначальное извлечение полных текстов из PDF благодаря специальному программному обеспечению «pdftotext» и дальнейшую обработку текста при помощи инструментов «Томита-парсера» от компании Яндекс [68]. «Томита-парсер» позволяет извлекать структурированные данные (факты) из текстов на естественном языке. Проектируя необходимые исходные грамматики для парсера, становится возможным извлечение метаданных из оцифрованных печатных материалов [79], которые распознаются парсером как факты внутри текста. Сами факты по итогу работы могут быть импортированы в целевую базу данных конструктора после соответствующего конфигурирования (рисунок 8).

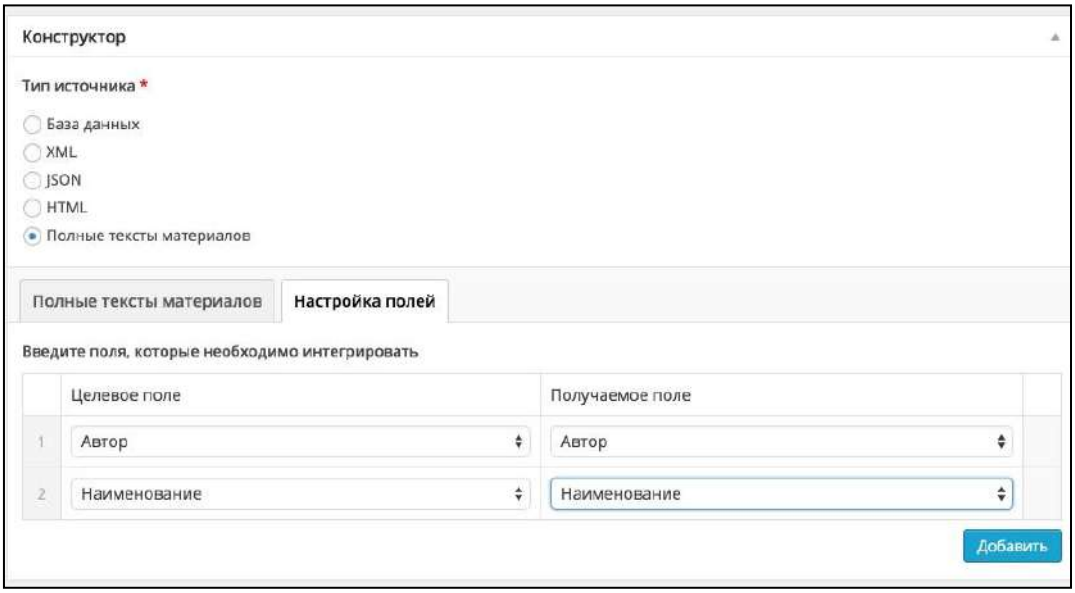


Рисунок 8. Настройка полей

Использование подобного метода интеграции полнотекстовых материалов позволяет расширить возможности упрощенного наполнения базы данных требуемой электронной библиотеки.

Для демонстрации функционирования конструктора правил интеграции полей был проведен эксперимент с извлечением мета-атрибутов о наименовании и полных имен авторов нескольких книг из электронной библиотеки им. Б. Н. Ельцина [58]. Доступа к SQL-серверу электронной библиотеки им. Б. Н. Ельцина

не было. Также эта электронная библиотека не имеет публично доступного и открытого интерфейса API. Таким образом, не осталось иных вариантов кроме синтаксического разбора DOM-модели HTML-страниц. Этот метод потребовал указания точных URL-адресов для интересуемых книг (Рисунок).

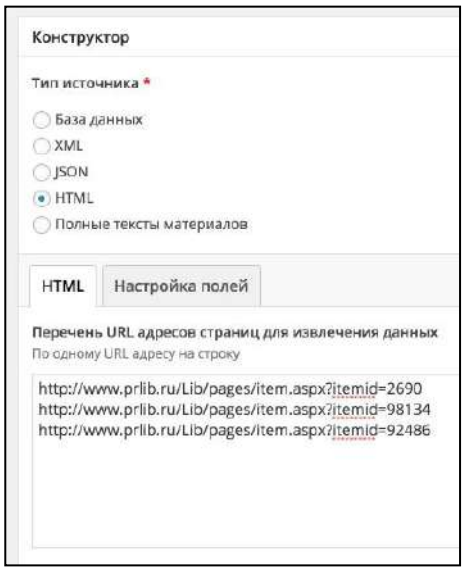


Рисунок 9. Пример работы с HTML в конструкторе

Для парсинга метаданных требуется разобрать HTML-код исходных страниц. Далее осуществляется выборка относительных элементов DOM-модели, содержащих метаданные. Пример поиска требуемых атрибутов показан на

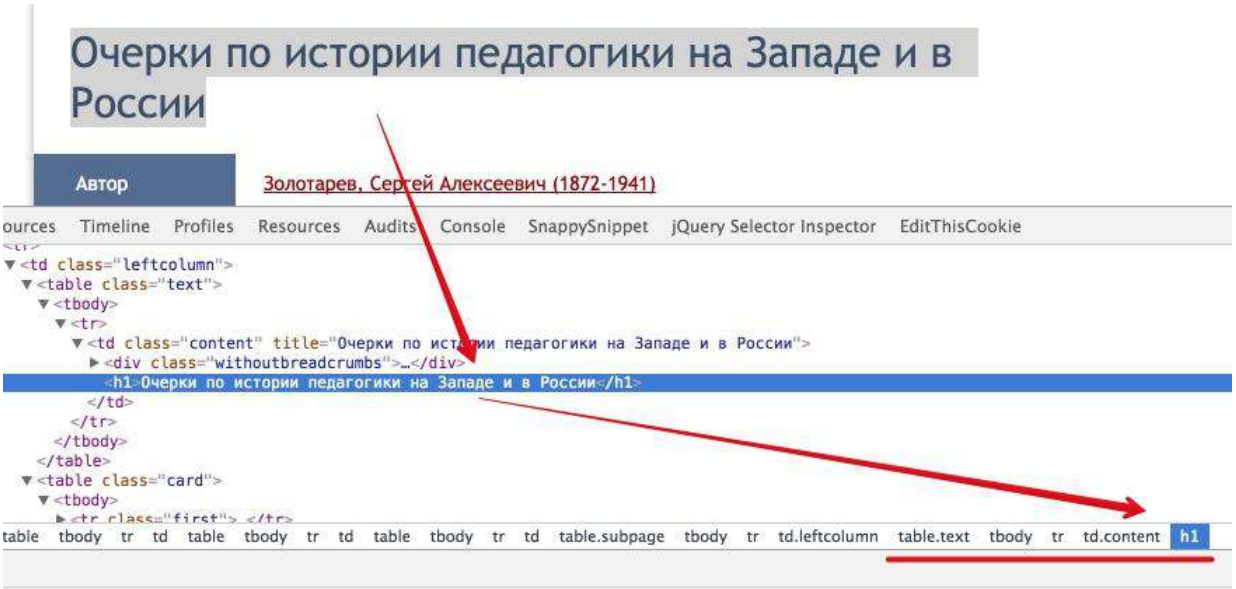


Рисунок .

Рисунок 10**.** Пример анализа кода HTML-страницы

После ручной фильтрации отобранных элементов HTML-разметки необходимо сконфигурировать правила, описывающие соответствие исходных и целевых полей в электронной библиотеке (Рисунок 9).

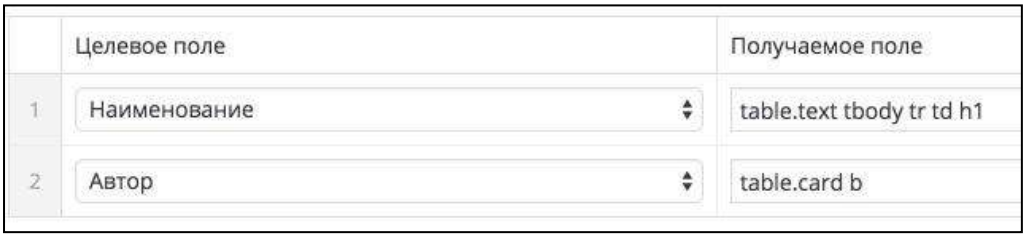


Рисунок 9. Правила добавления путей

Производя обход всего списка адресов страниц, содержащих получаемые поля, становится возможным повторное применение разработанных правил для всех страниц внешней библиотеки. Можно подать на вход конструктору карту сайта внешней библиотеки, и система обойдет все ссылки по очереди, извлечет требуемые поля и обновит их в исходной базе данных.

Спроектированный конструктор интеграции метаданных из разных источников позволяет извлекать требуемые поля из внешних электронных библиотек, если их данные хранятся в одном из машиночитаемых форматов (HTML, JSON, XML), либо при помощи прямой работы со внешней базой данных через SQL или API. Для каждого из источников требуется задать соответствия исходных и целевых полей и запустить конструктор для отработки всех источников.

Использование конструктора позволяет избавиться от задачи ручного копирования метаданных из разных внешних библиотек, однако не отменяет обязательной последующей работы редактора или администратора для проверки, корректировки и дополнения извлеченных мета-атрибутов.

Фрагменты исходного кода конструктора даны в Приложении.

2.4 Построение модели извлечения метаданных из полнотекстовых документов

В главе 1 данного диссертационного исследования была проведена оценка скорости роста объемов информации в сети интернет. Параллельно с экспоненциальным увеличением количества опубликованной информации одним из важнейших критериев для пользователей остается доступность этой информации для населения.

Электронные библиотеки остаются одной из наиболее важных точек входа к проверенной и качественной информации в интернете. Цифровые библиотеки обеспечивают доступ к полнотекстовым материалам с любого современного устройства с веб-браузером, в том числе со смартфонов и планшетов.

Очевидно, что для публикации материалов в электронной библиотеке данные материалы надо оцифровать. Оцифровка материалов сама по себе является сложным техническим процессом, который требует траты большого количества времени и усилий со стороны человека. Тем не менее сегодня уже существуют специализированные сканирующие устройства, позволяющие минимизировать ручной труд. Это специальные сканеры, внутрь которых помещаются раскрытые книги и сканеры самостоятельно переворачивают страницы при помощи направленных потоков воздуха, фотографируют их, а специальное программное обеспечение производит очистку изображений от шумов и дефектов.

Однако сам по себе процесс оцифровки является лишь частью подготовки электронного издания к публикации в электронной библиотеке. Неотъемлемой частью публикации является снабжение электронных материалов перечнем мета-атрибутов, которые в дальнейшем используются для поиска и рубрикации оцифрованных материалов. Именно подобный полный производственный цикл подготовки материалов позволяет наполнять электронные библиотеки качественной информацией.

Если оцифрованный материал уже имеется в интернете и опубликован в какой-то внешней электронной библиотеке, тогда метаданные можно попробовать извлечь из данной библиотеки при помощи API (при наличии) либо парсинга HTML-разметки страниц библиотеки.

В случае если материал не опубликован ни в одной из доступных электронных библиотек либо оцифровывается впервые, то метаданные потребуется извлекать редактору вручную. Это весьма трудоемкий и кропотливый процесс, требующий внимательности и проверок. Альтернативным способом может являться автоматизированное извлечение метаданных из полных текстов материалов, хранящихся на диске, при помощи методов синтаксического разбора текстов.

Для анализа и синтаксического разбора текста в данном диссертационном исследовании был использован Яндекс «Томита-парсер». Томита-парсер разбирает текст на естественном языке, учитывая синтаксис и морфологию [25] входящего текста.

Томита-парсер поставляется по лицензии Mozilla Public License (MPL) и имеет открытые исходные коды, выложенные на GitHUB. Для использования парсера нужно подготовить ряд конфигурационных файлов, описывающих механизмы извлечения метаданных из полных текстов на естественном языке:

– КС-грамматики (набор правил, описывающих синтаксическую структуру извлекаемых цепочек слов);

– газзетиры (словари с ключевыми словами для грамматик);

– файлы, описывающие факты (регулирует механизм преобразования грамматик в конкретные факты).

Проверка эффективности разработанной методики проводилась в рамках специально подготовленного эксперимента. Для теста был подготовлен массив из ста случайных книг из фонда одной из публичных электронных библиотек [70].

Эта выборка была отдана на вход разработанному алгоритму, использующему Томита-парсер для извлечения метаданных из полных текстов

материалов. Извлеченные материалы сравнивались с эталонными значениями – вручную введенными метаданными этих материалов в базе данных исходной электронной библиотеки.

Разработанные алгоритмы позволяют извлекать следующий набор метаданных:

– название материала;

– сведения об авторах;

– код ISBN (уникальный номер книжного издания);

– год публикации;

– место публикации;

– сведения об издателе;

– коды рубрикаторов (УДК, ББК, ГРНТИ).

Для анализа текстов были сформированы и использованы грамматики, приведенные в Таблица 2.

Таблица 2. Грамматики, разработанные для извлечения метаданных

Метаданные	Используемая грамматика
ISBN	S -> ('ISBN') (':') ('-') AnyWord<wfl="[0-9]{1,10}(-)?[0-9]{1,10}(-)?[0-9]{1,10}(-)?[0-9]{1,10}(-)?[0-9]{1}">; Isbn -> S interp (Material.Isbn);
Информация об издателе	PublisherDescr -> (Adj) 'издательство' | 'издательский' Noun; ForFact ->Word<h-reg1, gnc-agr[1], rt> (Word<gnc-agr[1]>*); CityOnly ->Word<gram="гео">; ForCity ->CityOnlyinterp (Material.PlaceOfPublish);

Метаданные	Используемая грамматика
	S -> (ForCity) PublisherDescrForFactinterp (Material.Publisher::not_norm); S -> (ForCity) PublisherDescrForFact<quoted>interp (Material.Publisher::not_norm);
Коды рубрикаторов	UDKStart -> 'удк' (':') ('-'); UDKDeskr ->AnyWord<wff=/[0-9]{1,5}(\.|-)?([0-9]{1,5})?((\.|-)?)([0-9]{1,5})?(\.|-)?([0-9]{1,5})?/>interp (Material.RubricsUDK) (','); UDK ->UDKStartUDKDeskr+; BBKStart -> 'ббк' (':') ('-'); BBKDeskr ->AnyWord<wff=/[0-9]{1,5}(\.|-)?([0-9]{1,5})?((\.|-)?)([0-9]{1,5})?(\.|-)?([0-9]{1,5})?/>interp (Material.RubricsBBK); BBK ->BBKStartBBKDeskr+; GrntiStart -> 'ггнти' (':') ('-'); GrntiDeskr ->AnyWord<wff=/[0-9]{1,5}(\.|-)?([0-9]{1,5})?((\.|-)?)([0-9]{1,5})?(\.|-)?([0-9]{1,5})?/>interp (Material.RubricsGrnti); Grnti ->GrntiStartGrntiDeskr+; S -> BBK | UDK | Grnti;
Дата и место публикации	CityOrOrg ->Word<gram="гео"> | "ран" interp (Material.PlaceOfPublish); S ->CityOrOrg (',') AnyWord<wfl="18[0-9]{2}|19[0-9]{2}|20[0-1][0-9]">interp (Material.YearOfPublish);

Метаданные	Используемая грамматика
Автор и наименование	Initial ->Word<wff=/[А-Я]\./>; Initials ->Initial<h-reg1>Initial<h-reg1>; FullName ->InitialsWord<gram="фам"> | Word<gram="фам">Initials | Word<gram="фам"> (',') Word<gram="имя">Word<gram="отч">; Person ->FullNameinterp (Material.Person::not_norm); Year -> (',') AnyWord<wfl="18[0-9]{2}|19[0-9]{2}|20[0-1][0-9]">interp (Material.YearOfPublish) ('.') EOSent ; FromStart ->AnyWord<fw, h-reg1>AnyWord*; MaterialName ->FromStartinterp (Material.Name::not_norm) ('/') Person; NotFromStart ->AnyWord<h-reg1>AnyWord*; MaterialName -> 'научный' 'издание' NotFromStartinterp (Material.Name::not_norm);

2.5 Анализ результатов извлечения метаданных из полнотекстовых документов

Создание представленных грамматик потребовало проведения отдельного исследования тестовой выборки полнотекстовых материалов. В ходе исследования ста случайных материалов общедоступной электронной библиотеки были выявлены некоторые повторяющиеся паттерны.

Алгоритмизация паттернов позволила уточнить методы и модели для повышения качества извлечения метаданных из полных текстов оцифрованных печатных материалов. В качестве наиболее значимых паттернов можно выявить следующие:

1. Все целевые метаданные опубликованы на первых или на последних трех страницах оцифрованного печатного материала.

Благодаря этому выводу можно существенно сократить временные затраты на автоматизированную обработку оцифрованных печатных материалов, так как все страницы внутри заданного диапазона содержат сам контент материала, а не его метаданные.

2. Наименование материала встречается в аннотации в двух возможных сочетаниях:

– «Наименование» / «Автор»;

– («Издание» или «Публикация») «Наименование».

3. Автор или группа авторов издания указываются близко к наименованию. Возможно указание как перед, так и после наименования материала. Имена авторов указываются в разных форматах, например: ФИО/ИОФ целиком, Инициалы и Фамилия, Фамилия и Инициалы.

4. Код ISBN обозначается путем проставления ключевого слова ISBN перед цифровой последовательностью, разделенной знаком «-».

5. Год и место публикации материала указываются рядом. В качестве места публикации могут выступать географические объекты (Москва, Россия) или наименовании организаций, например – РАН, институт и пр.

6. Сведения об издательстве начинаются с существительного «ИЗДАТЕЛЬСТВО» или прилагательного «ИЗДАТЕЛЬСКИЙ» с существительными, например, «ФИРМА» или «ДОМ»;

7. Коды рубрикаторов предваряются наименованием рубрикатора, например, «УДК» «КОДЫ».

Пример автоматического извлечения метаданных (фактов) приведен на Рисунок 10.

Material								
Name	Person	Isbn	YearOfPublish	PlaceOfPublish	Publisher	RubricsUDK	RubricsBBK	RubricsGrnti
						7.0		
							85	
			2009	РАН				
				Москва	Восточная литература			
Николаева Н. С. Образы Японии : очерки и заметки	Н. С. Николаева							
		978-5-02-036405-9						
		978-5-02-036405-9						
ISBN 978-5-02-036405-9 ©	Николаева Н. С.							
		978-5-02-036405-9						
			2009	РАН				
		710-794						

Рисунок 10. Пример извлечения фактов из полного текста книги

Приведенный пример показывает, что извлеченные метаданные требуют дополнительной программной обработки. Факты часто дублируются (если встречаются в книге несколько раз), иногда в наименование материала попадает ФИО автора и т. д. При тестировании алгоритма на больших выборках возможно появление дополнительных неточностей, что решается доработкой алгоритмов извлечения фактов, уточнения грамматик и последующей программной обработкой.

В целом результаты эксперимента подтверждают гипотезу о возможности автоматизированного извлечения мета-атрибутов из полнотекстовых материалов, сохраненных на диске. Вместе с тем очевидно, что использовать извлеченные

метаданные без дополнительной очистки и обработки редактором или администратором библиотеки не является корректным.

Результаты работы Томита-парсера могут быть сохранены в виде обычного текста или XML. Далее этот текст или разметка может быть загружена в целевую библиотеку, внутри которой следует осуществить дополнительную программную корректировку метаданных.

Дополнительная корректировка позволяет привести данные к общему виду, удалить дубли, попробовать восстановить обрывочные данные и исправить синтаксические ошибки, возникшие в ходе оптического распознавания текста.

2.6 Исследование моделей для повышения качества извлечения метаданных

После результатов извлечения метаданных из 100 полнотекстовых материалов публичной электронной библиотеки было принято решение увеличить выборку до 10 000 материалов. Методика исследования осталась прежней – автоматизировано извлеченные метаданные сравнивались с эталонными данными из базы данных.

После проведения экспериментов были получены следующие результаты, представленные в Таблица 2.

Таблица 2. Корректность извлечения метаданных из тестовой выборки материалов

Поле	Извлечено верно (%)	Извлечено неверно (%)	Требуется уточнение (%)
Наименование материала	76	21	3
Сведения об авторах	91	7	2
Код ISBN	98	0	2

Год публикации	89	10	1
Место публикации	84	12	4
Сведения об издателе	79	14	7
Коды рубрикаторов	90	1	9
Результаты в среднем	86,7	9,3	4

Средний показатель корректно извлеченных метаданных составляет 86,7%, еще 4% извлеченных фактов поддаются последующей корректировке и могут быть использованы после ее проведения.

В колонке «Требуется уточнение» показан процент данных, требующих корректировки для корректного извлечения. Например, в ходе работы парсера были обнаружены погрешности при оптическом распознавании текста (OCR).

Наибольшие проблемы наблюдаются с извлечением наименований материалов, которые не имеют четко утвержденной структуры, могут содержать любое количество символов и знаков препинания. Это делает невозможным создание однозначно корректных грамматик для извлечения сведений о наименовании.

Вторым по сложности для извлечения является поле сведений об издательстве и месте издания. Так же, как и с наименованием, для издательства не имеется четких правил написания, для которых можно разработать универсальные грамматики. Тем не менее при дополнительной обработке можно добиться уровня корректности извлечения выше 80% для сведений об издательстве и месте издания. Дополнительно может потребоваться подключение актуального словаря географических объектов и справочников организаций – это тоже может повысить процент извлеченных метаданных.

Коды ISBN, напротив, имеют четкую структуру написания. Так как сведения о кодах начинаются с ключевой аббревиатуры ISBN – написание соответствующей грамматики позволяет извлекать почти 100% корректных метаданных. Автор

выдвигает гипотезу, что подобных результатов можно добиться и с другими кодами, в частности кодами рубрикаторов (например, ГРНТИ, ББК и др.).

Благодаря извлечению номера ISBN становится возможным поиск (в том числе автоматизированный) сведений о материале в других электронных библиотеках, добавленных в конструктор. Также, зная код материала, можно запросить сведения об авторах через Google Books ISBN API и другие подобные сервисы.

Таким образом, автором был создан конструктор интеграции данных для электронной библиотеки, позволяющий объединять данные из других библиотек вне зависимости от используемого формата хранения метаданных, а также разработана модель извлечения метаданных из полных текстов материалов, с помощью которой возможна автоматизация извлечения метаданных в тех случаях, когда электронные материалы не сопровождаются метаданными.

Выводы по главе: автором рассмотрена концептуальная схема электронной библиотеки FRBR, описаны варианты связей в базе данных электронной библиотеки, предложена структурная схема электронной библиотеки, позволяющая обеспечить соответствие библиотеки предъявляемым требования, а также рассмотрена проблема интеграции данных из разных источников. Автором предложена модель, которая позволяет повысить качество и снизить трудовые затраты при интеграции данных из разных источников, связанные с различиями в хранении метаданных в разных электронных библиотеках. Предложенные алгоритмы позволяют автоматизировано извлекать атрибутивную информацию (метаданные), в том числе и из полных текстов, которые не сопровождаются метаданными в явном виде.

ЗАКЛЮЧЕНИЕ

В рамках исследования были обоснована модель и методика, направленные на улучшение качества интеграции цифровых информационных ресурсов из разных источников с учетом разных структур данных.

В главе 1 автор обратился к теоретическим основам создания электронных библиотек. Были представлены базовые определения основных терминов и описана краткая история появления и развития электронных библиотек. Были рассмотрены методы хранения данных в электронных библиотеках, описаны основные характеристики, процессы и отличия электронных библиотек от традиционных. Приведены изначальные принципы при проектировании автоматизированной библиотечно-информационной среды. Даны определения метаданных, стандартов и форматов их хранения, а также типизация информационных объектов.

В главе 2 автором была представлена практическая часть исследования. Была рассмотрена концептуальная схема электронной библиотеки FRBR, описаны варианты связей в базе данных электронной библиотеки, предложена структурная схема электронной библиотеки, позволяющая обеспечить соответствие библиотеки предъявляемым требованиям, проблема интеграции данных из разных источников. Автором предложена модель, которая позволяет повысить качество и снизить трудовые затраты при интеграции данных из разных источников, связанные с различиями в хранении метаданных в разных электронных библиотеках. Предложенные модели позволяют автоматизировано извлекать атрибутивную информацию (метаданные), в том числе и из полных текстов, которые не сопровождаются метаданными в явном виде.

В рамках исследования были рассмотрены особенности создания электронных библиотек, выделены основные характеристики, необходимые для успешного внедрения электронной библиотеки. В качестве изначальных

принципов при проектировании электронной библиотеки можно представить следующие принципы: однократность ввода исходных данных в библиотеку, выявление и использование определенного принципа хранения данных, доступ через сетевые ресурсы, обеспечение совместимости с внешними системами и сервисами, например, с другими электронными библиотеками. Структура электронных библиотек в целом аналогична структурам других информационных систем, однако имеет и свои особенности, в частности необходимость описания файлов и хранения их метаданных. При создании электронной библиотеки необходимо определить параметры проектирования электронной библиотеки, а также выбрать схему представления метаданных и учесть ее в архитектуре электронной библиотеки.

На основании проведенного анализа особенностей создания и характеристик электронных библиотек была разработана структура базы данных, отвечающая современным требованиям к электронной библиотеке. Предложенная структура подразумевает наполнение библиотеки как собственными ресурсами, так и интеграцию с другими базами данных, что позволяет обеспечить максимальную наполняемость библиотеки и, как следствие, ее универсальность и разнообразность. Представленная структура предполагает наличие возможности настройки таблиц, добавления и удаления полей, а также настройки доступа пользователей к библиотеке, что позволяет осуществить любые вариации доступа к ресурсам, а также обеспечить наиболее удобную для пользователя навигацию. Перечисленные возможности структуры базы данных направлены на выполнение одного из главных требований пользователя, а именно максимальное сокращение времени поиска информации, что позволит электронной библиотеке, использующей предлагаемую структуру, выгодно отличаться от иных электронных библиотек.

Были рассмотрены возможности интеграции внешних материалов в созданную электронную библиотеку с учетом разных форматов. Для обеспечения наиболее эффективного объединения ресурсов был спроектирован конструктор

полей интеграции данных, позволяющих объединять данные разных форматов через единый интерфейс. Спроектированный конструктор позволяет извлекать требуемые поля из внешних электронных библиотек, если их данные хранятся в одном из машиночитаемых форматов (HTML, JSON, XML), либо при помощи прямой работы со внешней базой данных через SQL или API. Для каждого из источников требуется задать соответствия исходных и целевых полей и запустить конструктор для отработки всех источников.

Исследована проблема интеграции материалов без метаданных, для решения которой предложено извлекать метаданные из полнотекстовых материалов. Алгоритм работы конструктора предполагает первоначальное извлечение полных текстов из PDF благодаря специальному программному обеспечению «pdftotext» и дальнейшую обработку текста при помощи инструментов «Томита-парсера» от компании Яндекс. «Томита-парсер» позволяет извлекать структурированные данные (факты) из текстов на естественном языке, которые по итогу работы могут быть импортированы в целевую базу данных конструктора после соответствующего конфигурирования.

Автором составлены и предложены специальные правила обработки оцифрованных печатных изданий для повышения эффективности автоматизированного поиска мета-атрибутов. Проведенные эксперименты продемонстрировали, что использование синтаксического разбора полнотекстовых печатных материалов на русском языке позволяет сократить усилия на наполнение электронной библиотеки метаданными, сопровождающими материалы. Эксперимент проводился в 2 этапа: на первом этапе в эксперименте участвовало 100 случайных материалов общедоступной электронной библиотеки, на втором этапе выборка увеличилась до 10 000 материалов. В ходе исследования ста случайных материалов общедоступной электронной библиотеки были выявлены некоторые повторяющиеся паттерны. Алгоритмизация паттернов позволила уточнить методы и модели для повышения качества извлечения метаданных из полных текстов оцифрованных печатных материалов.

По результатам экспериментов средний показатель корректно извлеченных метаданных составил 86,7%, еще 4% извлеченных фактов поддаются последующей корректировке и могут быть использованы после ее проведения.

Наибольшие проблемы возникают при извлечении информации, которая не имеет четко утвержденной структуры, может включать любое количество символов и знаков препинания. К такой информации относятся, например, наименования материалов и сведения об издательстве и месте издания. Отсутствие четкой структуры делает невозможным создание однозначно корректных грамматик для извлечения сведений о наименовании, издательстве и месте издания. Тем не менее при дополнительной обработке можно добиться уровня корректности извлечения выше 80% для сведений об издательстве и месте издания. Дополнительно может потребоваться подключение актуального словаря географических объектов и справочников организаций – это тоже может повысить процент извлеченных метаданных.

Коды ISBN, напротив, имеют четкую структуру написания. Так как сведения о кодах начинаются с ключевой аббревиатуры ISBN – написание соответствующей грамматики позволяет извлекать почти 100% корректных метаданных. Автор выдвигает гипотезу, что подобных результатов можно добиться и с другими кодами, в частности кодами рубрикаторов (например, ГРНТИ, ББК и др.).

Несмотря на полученные положительные результаты экспериментов, работа редактора или администратора библиотеки является обязательной для проверки, корректировки и утверждения автоматизировано извлеченных метаданных.

Процент успешного извлечения метаданных из полных текстов можно увеличить благодаря улучшению качества оптического распознавая печатных материалов, а также улучшению КС-грамматик и газзетиров (словарей).

Результаты исследования получили практическое применение. Методики, разработанные в рамках данной работы, используются в управлении библиотечным фондом электронной библиотеки Московского педагогического государственного университета. Разработанный конструктор позволил объединить

имеющиеся оцифрованные материалы для электронной библиотеки Московского педагогического государственного университета.

Отдельные модули интеграции данных и разработанный конструктор позволил автоматизировать управление библиотечным фондом Московского городского педагогического университета в части наполнения электронной библиотеки метаданными.

Исходный код конструктора правил интеграции информации из распределенных источников выложен в открытый репозиторий по лицензии GNU General Public License (универсальная общественная лицензия GNU).

Автором зарегистрированы две программы для ЭВМ:

– № 2012619529 – «Система управления контентом электронной библиотеки», дата регистрации 22.10.2012 (совместно с Шабановым Б. М., вклад автора – постановка задачи),

– № 2019661660 – «Конструктор правил интеграции данных для электронных библиотек», дата регистрации 05.09.2019 (без соавторов).

СПИСОК ЛИТЕРАТУРЫ

1. ГОСТ 34.601-90. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания : изд. офиц. : нац. стандарт : дата введения 1992-01-01. – Москва : Стандартинформ, 2009. – 5 с. – (Система стандартов по информации, библиотечному и издательскому делу).
2. ГОСТ Р 7.0.96–2016. Электронные библиотеки. Основные виды. Структура. Технология формирования : изд. офиц. : нац. стандарт : введен впервые : дата введения 2017-07-01. – Москва : Стандартинформ, 2016. – III, 13 с. – (Система стандартов по информации, библиотечному и издательскому делу).
3. Абросимов А. Г. Электронные библиотеки научных и образовательных ресурсов : учебно-методическое пособие / А. Г. Абросимов, Ю. И. Лазарева. – Казань : КГУ, 2008. – 78 с.
4. Авдеева Н. В. Национальные электронные библиотеки разных стран: реальность и перспективы / Н. В. Авдеева, И. В. Сусь // Информационные ресурсы России. – 2016. – № 2 (150). – С. 15–19.
5. Авторское право и библиотеки : руководство для библиотечных и информационных работников / Я. Л. Шрайберг [и др.]. – Москва : ГПНТБ России, 2007. – 47 с.
6. Анищенко Л. Н. Формирование и развитие системы электронных образовательных и научных ресурсов вузовской библиотеки // Научные и технические библиотеки. – 2016. – № 2. – С. 25–32.
7. Антопольский А. Б. Информационные ресурсы России // Научные и технические библиотеки. – 2000. – №1. – С. 27–33.

8. Антопольский А. Б. Системы метаданных в электронных библиотеках // Библиотеки и ассоциации в меняющемся мире: новые технологии и новые формы сотрудничества : материалы 8-й Междунар. конф. «Крым-2001». – Москва : [б.и.], 2001. – Т. 1. – С. 287–298.

9. Антопольский А. Б. Электронные библиотеки: принципы создания : научно-методическое пособие / А. Б. Антопольский, Т. В. Майстрович. – Москва : Либерея-Бибинформ, 2007. – 283 с. – (Библиотекарь и время. XXI век ; № 56).

10. Байдош Дж. Электронные ресурсы научно-технической информации в Библиотеке Конгресса США // Научные и технические библиотеки. – 2000. – № 11. – С. 58–76 ; № 12. – С. 54–76.

11. Бахмин А. В. Технические аспекты электронной доставки документов во ВГБИЛ // Библиотеки и ассоциации в меняющемся мире: новые технологии и новые формы сотрудничества : материалы 7-й Междунар. конф. «Крым-2000». – Москва : [б.и.], 2000. – Т. 2. – С. 449–451.

12. «Библиотечное дело, информационные системы и образование в США» – одиннадцатое профессиональное библиотечно-информационное мероприятие / Я. Л. Шрайберг, К. А. Колосов, М. В. Гончаров, Н. А. Каширина // Научные и технические библиотеки. – 2009. – № 9. – С. 83–94.

13. Бюлент И. Право на информацию: возможна ли его реализация в развивающихся странах? / Илмаз Бюлент // Научные и технические библиотеки. – 1999. – № 9. – С. 4–11.

14. Вебер Х. Оцифровка как метод обеспечения сохранности? / Х. Вебер, М. Дерр ; пер. с англ. А. И. Земскова ; науч. ред. д-р техн. наук Я. Л. Шрайберг. – Москва : ГПНТБ России, 1999. – 48 с.

15. Вегнер Б. Проект ЭЙЛЕР – интегрированный доступ к библиотечным каталогам и математической информации в Интернете // Научные и технические библиотеки. – 2001. – № 2. – С. 75–81.

16. Вислый А. И. Электронные библиотеки России. Проблемы формирования и использования // Библиотеки и ассоциации в меняющемся мире: новые технологии и новые формы сотрудничества : материалы 8-й Междунар. конф. «Крым-2001». – Москва : [б.и.], 2001. – Т. 1. – С. 298–302.

17. Воройский Ф. С. Организационно-технологические принципы сохранения машиночитаемых ресурсов автоматизированных библиотечно-информационных систем // Библиотеки и ассоциации в меняющемся мире: новые технологии и новые формы сотрудничества: материалы 7-й Междунар. конф. «Крым-2000». – Москва : [б.и.], 2000. – Т. 1. – С. 146–151.

18. Воройский Ф. С. Систематизированный толковый словарь по информатике. Вводный курс по информатике и вычислительной технике в терминах. – Москва : Либерея, 1998. – 375 с. – (Приложение к журналу «Библиотека» ; ч. 3).

19. Гончаров М. В. Введение в Интернет : учебное пособие : [в 9 ч.] / М. В. Гончаров, Я. Л. Шрайберг ; под общ. науч. ред. Я. Л. Шрайберга. – Москва : ГПНТБ России, 2000–2001. – 9 ч.

20. Гончаров М. В. Интернет/Интранет-технологии // Учебно-методические материалы / Моск. гос. ун-т культуры и искусств. Каф. информ. технологий и электрон. б-к ; науч. рук. Я. Л. Шрайберг. – Москва[б.и.], 2009. – Вып. 4. – С. 77–81.

21. Гончаров М. В. Информационные технологии. Ч. 3. Интернет/интранет-технологии : учебное пособие // Учебно-методические материалы / Моск. гос. ун-т культуры и искусств. Каф. информ. тех-нологий и электрон. б-к ; науч. рук. Я. Л. Шрайберг. – Москва : [б.и.], 2009. – Вып. 4. – С. 108–115.

22. Евстигнеева Г. А. Электронная информация – электронная библиотека. (Международный семинар в г. Пущине) : [краткое сообщение] / Г. А. Евстигнеева, А. И. Земсков // Научные и технические библиотеки. – 2000. – № 6. – С. 46–52.

23. Елизаров А.М. Свободно распространяемые системы управления электронными научными журналами и технологии электронных библиотек / А. М. Елизаров, Д. С. Зуев, Е. К. Липачёв // Электронные библиотеки: перспективные методы и технологии, электронные коллекции : тр. XV Всерос. науч. конф. RCDL'2013. – Ярославль : ЯрГУ, 2013. – С. 227–236.

24. Елисина Е. Ю. Электронные услуги библиотек . – Санкт-Петербург : Профессия, 2010. – 302 , [1] с. – (Библиотека).

25. Зайцева Е. М. Лингвистическое обеспечение автоматизированных библиотечно-информационных систем // Учебно-методические материалы / Моск. гос. ун-т культуры и искусств. Каф. информ. технологий и электрон. б-к ; науч. рук. Я. Л. Шрайберг. – Москва : [б.и.], 2009. – Вып. 4. – С. 15–22.

26. Земсков А. И. Авторское право на электронные документы в библиотеках // Учебно-методические материалы / Моск. гос. ун-т культуры и искусств. Каф. информ. технологий и электрон. б-к ; науч. рук. Я. Л. Шрайберг. – Москва : [б.и.], 2009. – Вып. 4. – С. 137–146.

27. Земсков А. И. К проекту Программы «Российские электронные библиотеки» // Научные и технические библиотеки. – 2000. – №3. – С. 4–10.

28. Земсков А. И. Конкретные модели и проекты открытого доступа / А. И. Земсков, Я. Л. Шрайберг // Научные и технические библиотеки. – 2008. – №7. – С. 34–44.

29. Земсков А. И. Системы открытого доступа к информации: причины и история возникновения / А. И. Земсков, Я. Л. Шрайберг // Научные и технические библиотеки. – 2008. – № 4. – С. 16–29.

30. Земсков А. И. Социальное разделение, вызванное электронными библиотеками // Библиотечное дело – 2001: Российские библиотеки в мировом информационном и интеллектуальном пространстве : тез. докл. 6-й Междунар. науч. конф., Москва, 26–27 апр. 2001 г. / Моск. гос. ун-т культуры и искусств. – Москва : МГУКИ, 2001. – Ч. 1. – С. 18–19.

31. Земсков А. И. Электронная информация и электронные ресурсы: публикации и документы, фонды и библиотеки / А. И. Земсков, Я. Л. Шрайберг. – Москва : ФАИР, 2007. – 527, [1] с. – (Специальный издательский проект для библиотек).

32. Земсков А. И. Электронные библиотеки // Учебно-методические материалы / Моск. гос. ун-т культуры и искусств. Каф. информ. технологий и электрон. б-к ; науч. рук. Я. Л. Шрайберг. – Москва, 2009. – Вып. 4. – С. 82–91.

33. Земсков А. И. Электронные библиотеки : учебник для студ. вузов, обуч. по спец. 052700 "Библ.-информ. деятельность" / А. И. Земсков, Я. Л. Шрайберг ; отв. ред. О. Бородин. – Москва : Либерея, 2003. – 351 с. – (Альманах "Приложение к журналу "Библиотека" ; 2-е полугодие 2003 г.).

34. Земсков А. И. Электронные библиотеки : учебное пособие / А. И. Земсков, Я. Л. Шрайберг ; Моск. гос. ун-т культуры и искусств, Гос. публ. науч.-техн. б-ка России. – Москва : [б.и.], 2001. – 91 с.

35. Земсков А. И. Электронные библиотеки : учебное пособие для студ. ун-тов и вузов культуры и искусств и др. учеб. заведений / А. И. Земсков, Я. Л. Шрайберг ; Моск. гос. ун-т культуры и искусств. – 3-е изд., испр. и доп. – Москва : ГПНТБ России, 2004. – 130 с.

36. Земсков А. И. Электронные библиотеки и общественная активность // Научные и технические библиотеки. – 2002. – № 3. – С. 14–17.

37. Земсков А. И. Электронные публикации // Учебно-методические материалы / Моск. гос. ун-т культуры и искусств. Каф. информ. технологий и электрон. б-к ; науч. рук. Я. Л. Шрайберг. – Москва : [б.и.], 2009. – Вып. 4. – С. 92–96.

38. Иванов В. С. Конференция «Библиотеки и образование»: итоги и перспективы / В. С. Иванов, Я. Л. Шрайберг // Библиотеки и образование : сб. материалов 1-й Междунар. конф., Ярославль, 19–22 апр. 2005 г. – Ярославль : МУБиНТ, 2005. – С. 7–10.

39. Информационно-психологическая безопасность : (Определение и анализ предметной области) / Г. Л. Смолян, Г. М. Зараковский, В. М. Розин, А. Е. Войскунский ; Институт системного анализа РАН. – Москва : ИСА, 1997. – 52 с.

40. Каптерев А. И. Концепция информатизации университета // Научные и технические библиотеки. – 2000. – № 4. – С. 10–16.

41. Каракозов С. Д. Ориентиры развития цифровой образовательной среды Московского педагогического государственного университета / С. Д. Каракозов, Р. С. Сулейманов, А. Ю. Уваров // Наука и школа. – 2014. – № 6. – С. 69–83.

42. Каракозов С. Д. Техническая политика и этапы развития цифровой образовательной среды МПГУ / С. Д. Каракозов, Р. С. Сулейманов, А. Ю. Уваров // Наука и школа. – 2015. – № 1. – С. 17–27.

43. Каспарова Н. Н. Библиографическое описание электронных ресурсов в России: национальные аспекты и международный опыт // Научные и технические библиотеки. – 2000. – № 3. – С. 14–16.

44. Колосов К. А. Корпоративные библиотечные технологии // Учебно-методические материалы / Моск. гос. ун-т культуры и искусств. Каф. информ. технологий и электрон. б-к ; науч. рук. Я. Л. Шрайберг. – Москва : [б.и.], 2009. – Вып. 4. – С. 67–76.

45. Колосов К. А. Языки разметки HTML и XML // Учебно-методические материалы / Моск. гос. ун-т культуры и искусств. Каф. информ. технологий и электрон. б-к ; науч. рук. Я. Л. Шрайберг. – Москва : [б.и.], 2009. – Вып. 4. – С. 103–107.

46. Колосов К. А. WWW-серверы // Учебно-методические материалы / Моск. гос. ун-т культуры и искусств. Каф. информ. технологий и электрон. б-к ; науч. рук. Я. Л. Шрайберг. – Москва : [б.и.], 2009. – Вып. 4. – С. 97–102.

47. Кузнецова Т. Я. Сетевое взаимодействие как базовый фактор инновационного развития библиотечного образования // Научные и технические библиотеки. – 2018. – № 4 . – С. 84–97.

48. Кузьмин Е. И. Библиотечная Россия на рубеже тысячелетий. – Москва : Либерея, 1999. – 223 с.

49. Лопатина Н. В. Библиотечная профессия в информационном обществе: разрушение или развитие // Научно-техническая информация. Сер.1. Организация и методика информационной работы. – 2014. – № 5. – С. 19–23.

50. Лютецкий В. М. Автоматическая систематизация библиографических записей, достижения и проблемы : [видеозапись выступления на XXIV Ежегод. конф. РБА, Тула, 11-17 мая 2019] // Канал Центра ЛИБНЕТ : [канал пользователя видеохостинга YouTube]. – 31 мая 2019. – (24 мин. 06 с.). – URL: https://www.youtube.com/watch?v=pk5RxkSyaic (дата обращения: 15.03.2020).

51. Мазурицкий А. М. Идеология и библиотеки // Вестник Московского государственного университета культуры и искусств. – 2015. – № 2 (64). – С. 182–186.

52. Манилова Т. Л. Информационные ресурсы российских библиотек: социальный аспект // Научные и технические библиотеки. – 2001. – № 8. – С. 12–16.

53. Метаописания и каталогизация научно-информационных ресурсов РАН / А. О. Еркимбаев, А. Б. Жижченко, В. Ю. Зицерман [и др.] // Программные продукты и системы. – 2012. – № 3. – С. 117–123.

54. Морган Э. Электронные книги, библиотеки и право собственности // Научные и технические библиотеки. – 2001. – № 8. – С. 27–35.

55. МСЦ РАН (Межведомственный суперкомпьютерный центр Российской академии наук) – филиал ФГУ ФНЦ НИИСИ РАН : официальный сайт. – Москва, 1996 – . – URL: http://www.jscc.ru/ (дата обращения: 22.08.2018).

56. Мэррей Р. Компоненты цифровой библиотеки и их взаимодействие // Научные и технические библиотеки. – 2000. – № 6. – С. 56–68.

57. Предметно-ориентированные и междисциплинарные цифровые коллекции в электронном пространстве знаний / А. Н. Сотников, И. Н. Соболевская, С. А. Кириллов, И. Н. Чередниченко // Научный сервис в сети Интернет : тр. XX Всерос. науч. конф., Новороссийск, 17-22 сент. 2018. – Москва : ИПМ им. М. В. Келдыша, 2018. – № 20. – С. 448–453. – URL: http://keldysh.ru/abrau/2018/theses/52.pdf doi:10.20948/abrau-2018-52 (дата обращения: 15.03.2020).

58. Президентская библиотека имени Б. Н. Ельцина : сайт. – Санкт-Петербург, 2009 – . – URL: https://www.prlib.ru (дата обращения: 16.08.2016).

59. Принципы построения и формирования электронной библиотеки "Научное наследие России" / Н. Е. Калёнов, Г. И. Савин, В. А. Серебряков, А. Н. Сотников // Программные продукты и системы. – 2012. – № 4. – С. 30–40.

60. Российский коммуникативный формат представления библиографических записей в машиночитаемой форме : (рос. версия UNIMARC) / М-во культуры Рос. Федерации, Рос. библ. ассоц. // Национальная Служба развития системы форматов RUSMARC : [сайт].– Санкт-Петербург, [2001-2019]. – URL: http://rusmarc.ru/rusmarc/format.html (дата обращения: 16.03.2020). – Дата обновления: 13.12.2019.

61. Сайфутдинов Р. А. Электронная библиотека как средство эффективности компьютерного обучения / Р. А. Сайфутдинов, В. А. Лукьянов // Прикладные информационные системы : сб. науч. тр. Второй Всерос. науч.-практ. конф., Ульяновск , 25 мая–07 июня 2015 г. – Ульяновск : УГТУ, 2015. – С. 51–56.

62. Соколинский К. Е. Функции интегративного поиска вузовских библиотечных порталов, построенных на основе J-ИРБИС 2.0. / К. Е. Соколинский, Е. В. Крылова // Научные и технические библиотеки. – 2017. – № 11. – С. 82–90.

63. Соколова Ю. В. Роль информационно-библиотечной службы в электронном обучении / Ю. В. Соколова, Я. Л. Шрайберг // Информационные ресурсы и сервисы открытого образования : сб. материалов 3-й Междунар. науч.-практ. конф. "Библиотеки и образование", Кострома, 24–27 апр. 2007 г. – Ярославль : МУБиНТ, 2007. – С. 186–189.

64. Сулейманов Р. С. Извлечение метаданных из полнотекстовых электронных русскоязычных изданий при помощи томита-парсера // Программные продукты и системы. – 2016. – № 4. – С. 58–62.

65. Сулейманов Р. С. Сбор библиотечной информации из распределенных электронных источников при помощи конструктора правил интеграции данных // Информационные ресурсы России. – 2016. – № 6. – С. 23–26.

66. Сулейманов Р. С. Современные подходы к интеграции данных в электронных библиотеках // Информационные ресурсы России. – 2019. – № 6. – С. 13–16.

67. Сулейманов Р. С. Социальная сеть РАН – единое информационное пространство для ученых // Программные продукты и системы. – 2012. – № 4. – С. 46–49.

68. Томита-парсер // Технологии Яндекса. – Москва, 2014-2020. – URL: https://yandex.ru/dev/tomita/ (дата обращения: 16.03.2020).

69. Тютюнник В. М. Анализ данных и модель информационных процессов для формирования прикладных информационных систем // Промышленные АСУ и контроллеры. – 2019. – № 4. – С. 19–29.

70. Хаависто Т. Лицензирование и публичные библиотеки // Научные и технические библиотеки. – 2001. – № 3. – С. 107–112.

71. Хи Гвон Ю. «Визуальные сокровища» – проект Нью-Йоркской публичной библиотеки. Оцифровка русских визуальных ресурсов // Научные и технические библиотеки. – 2001. – № 8. – С. 50–55.

72. Цветков В. Я. Информационная угроза – СПАМ / В. Я. Цветков, С. В. Булгаков // Известия высших учебных заведений. Геодезия и аэрофотосъемка. – 2004. – № 5. – С. 118–130. – Электрон. копия доступна на сайте Науч. электрон. б-ки eLIBRARY.RU. URL: https://www.elibrary.ru/item.asp?id=25226168 (дата обращения: 15.03.2020). – Доступ после регистрации.

73. Цветкова В. А. Общество знаний и российская информационная инфраструктура / В. А. Цветкова, И. И. Родионов // Информационные ресурсы России. – 2019. – № 2. – С.9–13.

74. Шрайберг Я. Л. Авторское право и открытый доступ. Достоинства и недостатки модели открытого доступа / Я. Л. Шрайберг, А. И. Земсков // Научные и технические библиотеки. – 2008. – № 6. – С. 31–41.

75. Шрайберг Я. Л. Библиотеки в условиях правовой и технологической эволюции процессов общественного развития : ежегод. докл. конф. «Крым», год 2008. – Судак ; Москва : ГПНТБ России, 2008. – 56 с.

76. Шрайберг Я. Л. Библиотеки в электронной среде и вызовы современного общества : ежегод. докл. конф. «Крым», год 2009 // Научные и технические библиотеки. – 2010. – № 1. – С. 7–46.

77. Шрайберг Я. Л. Библиотеки, создающие будущее / Я. Л. Шрайберг, Е. В. Линдеман, Е. М. Зайцева // Университетская книга. – 2009. – № 10. – С. 20–24.

78. Шрайберг Я. Л. Интеграция библиотек в развивающееся информационное общество: что нас ждет впереди? : ежегод. докл. конф. «Крым», год 2012. – Москва : ГПНТБ России, 2012. – 63 с.

79. Шрайберг Я. Л. Использование печатных и электронных источников в фондах учебных и научных библиотек // Электронные ресурсы и международный информационный обмен: Восток–Запад : тр. 9-го Междунар. семинара, Вашингтон [и др.] , 2007. – Вашингтон [и др.] : [б.и.], 2007. – С. 52–53.

80. Шрайберг Я. Л. Как создать свой Web-сервер / Я. Л. Шрайберг, М. В. Гончаров. – Москва : Либерея, 2000. – 64 с. – (С компьютером на «ты» : справ. пособие для б-к по информац. технологиям и Интернет ; 2000, вып. 4).

81. Шрайберг Я. Л. Корпоративные и национальные проекты Открытого доступа / Я. Л. Шрайберг, А. И. Земсков // Научные и технические библиотеки. – 2008. – № 8. – С. 5–23.

82. Шрайберг Я. Л. Международные машиночитаемые форматы и корпоративные системы / Я. Л. Шрайберг, Э. Ш. Лобанова // Российское библиографоведение: итоги и перспективы : сб. науч. статей / сост. и науч. ред. Т. Ф. Лиховид. – Москва : ФАИР-ПРЕСС, 2006. – С. 644–678.

83. Шрайберг Я. Л. Модели открытого доступа: история, виды, особенности, терминология / Я. Л. Шрайберг, А. И. Земсков // Научные и технические библиотеки. – 2008. – № 5. – С. 68–79.

84. Шрайберг Я. Л. Права интеллектуальной собственности в России и кто ими владеет: источники информации // Электронные ресурсы и международный информационный обмен: Восток-Запад : тр. 8-го Междунар. семинара , Нью-Хэвен [и др.], 2006. – Москва : МБИАЦ : ГПНТБ России : [б.и.], 2006. – Т. 1 : Доклады. – С. 23–34.

85. Шрайберг Я. Л. Состояние Открытого доступа на библиотечно-информационном пространстве России и СНГ // Научные и технические библиотеки. – 2009. – № 11. – С. 29–38.

86. Шрайберг Я. Л. Сравнительный анализ деятельности и перспектив развития отечественных и зарубежных библиотечных консорциумов. Ч. 1 / Я. Л. Шрайберг, Е. В. Линдеман // Научные и технические библиотеки. – 2005. – № 7. – С. 5–15.

87. Шрайберг Я. Л. E-learning в России: нужны ли библиотеки? // Электронные ресурсы и международный информационный обмен: Восток–Запад : тр. 9-го Междунар. семинара, Вашингтон [и др.] , 2007. – Вашингтон [и др.] : [б. и.], 2007. – С. 76–79.

88. Электронная библиотека «Научное наследие России»: состояние и перспективы развития / Н. Е. Каленов, К. П. Погорелко, В. А. Серебряков, А. Н. Сотников. – DOI: 10.20948/abrau-2016-27 // Научный сервис в сети Интернет : труды XVIII Всерос. науч. конф., Новороссийск, 19-24 сент. 2016. – Москва : ИПМ им. М.В.Келдыша, 2016. – С. 148–151.

89. Эшкрофт Л. Изучение использования электронных журналов / Л. Эшкрофт, К. Лэнгдон // Научные и технические библиотеки. – 2000. – № 5. – С. 88–94.

90. Global - 2020 Forecast Highlights // Cisco : [official site]. – San Jose, California, 2016. – URL: https://www.cisco.com/c/dam/m/en_us/solutions/service-provider/vni-forecast-highlights/pdf/Global_2020_Forecast_Highli... (дата обращения: 19.12.2019).

91. Europeana Collections : european digital library : website. – Netherlands : EC, 2008 – . – URL: http://www.europeana.eu/portal/en (дата обращения: 22.08.2016).

92. Evolution of the Internet and its Cores / Guo-Qing Zhang, Guo-Qiang Zhang, Qing-Feng Yang [et al.]. – DOI: 10.1088/1367-2630/10/12/123027 // New Journal of Physics : the open-access journal for physics. – 2008. – N 10. –

URL: https://iopscience.iop.org/article/10.1088/1367-2630/10/12/123027/pdf (дата обращения: 13.03.2020).

93. Savin G. I. Comparative analysis of solutions for full-text search in digital libraries / G. I. Savin, A. N Sotnikov, R. S. Suleymanov // Innovative information technologies : proc. of the 3-rd Intern. sci.-practical conf., Prague, 21–25 Apr. 2014. – Moscow : HSE, 2014. – Part 2 : Innovative information technologies in science. – P. 624–629.

94. Tillett B. B. What is FRBR? A conceptual model for the bibliographic universe / Barbara B. Tillett ; Cataloging distribution service. – Washington, D. C.: Libr. Congr., 2004. – 8 p. – Электрон. копия доступна на сайте Library of Congress. URL: https://www.loc.gov/cds/downloads/FRBR.PDF (дата обращения: 04.02.2016).

95. Worldwide Internet of Things Forecast, 2019-2023 / Carrie MacGillivray, Marcus Torchia, Ashutosh Bisht [et al.]. – Doc # US45373120 // International Data Corporation (IDC) : [official site]. – [Framingham, Mass.], Sept. 2019. – URL: https://www.idc.com/getdoc.jsp?containerId=US45373120 (дата обращения: 15.03.2020). – Доступ платный.

96. What is the Document Object Model? / ed. Jonathan Robie // Document Object Model (DOM) Level 1 Specification : Version 1.0 : W3C Recommendation 1 Oct. 1998. – 1998. – P. 9–14. – Документ доступен на сайте World Wide Web Consortium. URL: https://www.w3.org/TR/1998/REC-DOM-Level-1-19981001/DOM.pdf (дата обращения: 22.06.2017).

ПРИЛОЖЕНИЯ

ИСХОДНЫЕ КОДЫ

Грамматика для извлечения ISBN:

```
S -> ('ISBN') (':') ('-') AnyWord<wfl="[0-9]{1,10}(-)?[0-9]{1,10}(-)?[0-9]{1,10}(-)?[0-9]{1,10}(-)?[0-9]{1}">;
Isbn -> S interp (Material.Isbn);
```

Грамматика для извлечения информации об издателе:

```
PublisherDescr -> (Adj) 'издательство' | 'издательский' Noun;
ForFact -> Word<h-reg1, gnc-agr[1], rt> (Word<gnc-agr[1]>*);

CityOnly -> Word<gram="гео">;
ForCity -> CityOnly interp (Material.PlaceOfPublish);

S -> (ForCity) PublisherDescr ForFact interp (Material.Publisher::not_norm);
S -> (ForCity) PublisherDescr ForFact<quoted> interp (Material.Publisher::not_norm);
```

Грамматика для извлечения информации о кодах рубрикаторов:

```
UDKStart -> 'удк' (':') ('-');
UDKDeskr -> AnyWord<wff=/[0-9]{1,5}(\.|-)?([0-9]{1,5})?((\.|-)?)([0-9]{1,5})?(\.|-)?([0-9]{1,5})?/> interp (Material.RubricsUDK) (',');
UDK -> UDKStart UDKDeskr+;
```

```
BBKStart -> 'ббк' (':') ('-');
BBKDeskr -> AnyWord<wff=/[0-9]{1,5}(\.|-)?([0-9]{1,5})?((\.|-)?)([0-9]{1,5})?(\.|-)?([0-9]{1,5})?/> interp (Material.RubricsBBK);
BBK -> BBKStart BBKDeskr+;

GrntiStart -> 'грнти' (':') ('-');
GrntiDeskr -> AnyWord<wff=/[0-9]{1,5}(\.|-)?([0-9]{1,5})?((\.|-)?)([0-9]{1,5})?(\.|-)?([0-9]{1,5})?/> interp (Material.RubricsGrnti);
Grnti -> GrntiStart GrntiDeskr+;

S -> BBK | UDK | Grnti;
```

Грамматика для извлечения информации о дате и месте публикации:

```
CityOrOrg -> Word<gram="гео"> | "ран"  interp (Material.PlaceOfPublish);

S -> CityOrOrg (',') AnyWord<wfl="18[0-9]{2}|19[0-9]{2}|20[0-1][0-9]"> interp (Material.YearOfPublish);
```

Грамматика для извлечения информации об авторах и наименовании:

```
Initial -> Word<wff=/[А-Я]\./>;

Initials -> Initial<h-reg1> Initial<h-reg1>;

FullName -> Initials Word<gram="фам">
          | Word<gram="фам"> Initials
          | Word<gram="фам"> (',') Word<gram="имя"> Word<gram="отч">;
```

```
Person -> FullName interp (Material.Person::not_norm);

Year -> (',') AnyWord<wfl="18[0-9]{2}|19[0-9]{2}|20[0-1][0-9]"> interp (Material.YearOfPublish) ('.') EOSent ;

FromStart -> AnyWord<fw, h-reg1> AnyWord*;
MaterialName -> FromStart interp (Material.Name::not_norm) ('/') Person;

NotFromStart -> AnyWord<h-reg1> AnyWord*;
MaterialName -> 'научный' 'издание' NotFromStart interp (Material.Name::not_norm);
```

Фрагменты исходного кода электронной библиотеки:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="author" content="">
    <link rel="icon" href="./favicon.ico">

    <title><?php wp_title('',true,'right'); ?></title>

    <!--[if lt IE 9]>
      <script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
      <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
    <![endif]-->
```

```
<? wp_head(); ?>
<link rel='stylesheet' id='style-css' href='http://library.ruslan.cc/wp-content/themes/library/css/style.css' type='text/css' media='all' />
</head>

<body>
<div class="container pre-header">
<div class="row">
<div class="col-md-6">
<div class="row">
<div class="col-sm-6 logo">
<a href='/'><img src="http://library.ruslan.cc/wp-content/themes/library/images/logo.png" alt="logo"></a>
</div>
<div class="col-sm-6 search"><form action="http://library.ruslan.cc/" method="get"><input type="text" name="s" class="form-control" placeholder="Поиск по библиотеке"></form></div>
</div>
</div>
<div class="col-md-6 links">
<a class="favorites" href="/favorites/">Избранное</a>
<?
if (is_user_logged_in()) {
?>
<a class="login" href="/login/">Личный кабинет</a>
```

```
                                        <a class="signin" href="<? echo wp_logout_url(); ?>&redirect_to=/login/">Выйти</a>
                                    <?
                                } else {
                                    ?>
                                        <a class="login" href="/login/">Вход</a>
                                        <a class="signin" href="/registration/">Регистрация</a>
                                    <?
                                }
                                ?>
                            </div>
                        </div>
                    </div>
                </div>

                <?php
                    $uri = $_SERVER["REQUEST_URI"];
                    preg_match_all('/^\/([a-z]*)\//', $uri, $matches);
                    $address = isset($matches[0][0]) ? $matches[0][0] : '';
                ?>

                <header>
                    <div class="container">
                    <nav role="navigation" class="navbar navbar-default">
                    <div class="navbar-header">
                        <button type="button" data-target="#navbarCollapse" data-toggle="collapse" class="navbar-toggle">
```

```
                <span class="sr-only">Toggle navigation</span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                </button>
            </div>

            <div id="navbarCollapse" class="collapse navbar-collapse">
                <ul class="nav navbar-nav">
                    <li <?php if (!$address) echo 'class="active"'; ?>><a
href="http://library.ruslan.cc/"><span class="border">О библиотеке</span></a></li>
                    <li class="divider"></li>
                    <li <?php if ($address == '/updates/') echo 'class="active"';
?>><a href="/updates/">Последние поступления</a></li>
                    <li class="divider"></li>
                    <li <?php if ($address == '/kollektsii/') echo 'class="active"';
?>>
                        <a href="/kollektsii">
                            Коллекции
                        </a>
                        <!-- <ul class="dropdown-menu">
                            <li><a
href="/category/collections/estestvennoistoricheskie-
kollektsii/">Естественноисторические коллекции</a></li>
                            <li><a
href="/category/collections/grnti/">Рубрикатор ГРНТИ</a></li>
                        </ul> -->
                    </li>
                    <li class="divider"></li>
```

```
                        <li class="dropdown">
                            <a href="/katalog-fondov" class="dropdown-toggle" data-toggle="dropdown">
                                Каталог фондов
                                <b class="caret"></b>
                            </a>
                            <ul class="dropdown-menu">
                                <li><a href="/books">Книги / Монографии / Рукописи</a></li>
                                <li><a href="/videos">Видео-записи</a></li>
                                <li><a href="/shemyi-kartyi">Схемы / Карты</a></li>
                            </ul>
                        </li>
                        <li class="divider"></li>
                        <li <?php if ($address == '/uchenyie/') echo 'class="active"'; ?>><a href="/uchenyie/">Учёные</a></li>
                        <li class="divider"></li>
                        <li <?php if ($address == '/organizatsii/') echo 'class="active"'; ?>><a href="/organizatsii/">Организации</a></li>
                    </ul>
                </div>
                </nav>
                </div>
        </header>

<?php get_header(); ?>
<div class="container m30t">
        <div class="row">
```

```
            <div class="col-xs-12">
                <?php the_breadcrumb(); ?>
            </div>
        </div>
        <div class="row m30t">
            <?php if ( have_posts() ) : while ( have_posts() ) : the_post(); ?>
        <!-- <div class="col-xs-12 col-sm-4">
            <?php the_nav_menu(); ?>
        </div> -->
            <div class="col-xs-12 col-sm-12">
                <h1 class="deep-blue"><?php the_title();?></h1>
                <?php the_content();?>
            </div>
            <?php endwhile; endif; ?>
    </div>
</div>

<?php get_footer(); ?>

            <div class="container-fluid footer">
                <div class="container">
                    <div class="row">
                        <div class="col-sm-3 design"><img
src="http://library.ruslan.cc/wp-content/themes/library/images/dp.png"
alt="design"></div>
                        <div class="col-sm-6" style="text-align: center;">©
Электронная библиотека, 2007 — 2019 <br>Условия использования
материалов</div>
```

```
                    <div class="col-sm-3 favorites"><a href="/favorites/">Избранное</a></div>
                </div>
            </div>
        </div>
        <? wp_footer(); ?>
    </body>
</html>


<? get_header(); ?>
  <div class="container m30t">
        <div class="row">
            <div class="col-xs-12 col-md-8">
                <h1>Электронная библиотека</h1>
                <br>
                <br>
                <? echo get_new_royalslider(1); ?>

            </div>
            <div class="col-xs-12 col-md-4">
                <h1>Статистика библиотеки</h1>
                <br>
                <ul class="nav nav-pills nav-stacked">
                    <?
                        $count_posts = wp_count_posts();
                        $category = get_the_category(6139);
                        $categoryv = get_the_category(33);
                        $categoryk = get_the_category(47904);
```

```
                $count_posts2 = wp_count_posts('authors');
                $count_posts3 = wp_count_posts('organizations');
                $count_posts4 = wp_count_posts('sources');
            ?>
            <li role="presentation"><b></b><a href="#">Всего материалов <span class="badge"><?=$count_posts->publish;?></span></a></li>
            <li role="presentation"><a href="#">Печатных изданий <span class="badge"><?=$category[0]->category_count;?></span></a></li>
            <li role="presentation"><a href="#">Карт/схем/планов <span class="badge"><?=$categoryk[0]->category_count;?></span></a></li>
            <li role="presentation"><a href="#">Видео-записей <span class="badge"><?=$categoryv[0]->category_count;?></span></a></li>
            <li role="presentation"><a href="#">Количество персон <span class="badge"><?=$count_posts2->publish;?></span></a></li>
            <li role="presentation"><a href="#">Организаций <span class="badge"><?=$count_posts3->publish;?></span></a></li>
            <li role="presentation"><a href="#">Источников <span class="badge"><?=$count_posts4->publish;?></span></a></li>
        </ul>
        <br>
        <h3>Новые персоны</h3>
        <? echo get_new_royalslider(3); ?>

        <div class="row m30t social-buttons">
            <div class="col-xs-3"><center><a href="http://vkontakte.ru/mpgu_edu" target="_blank"><img src="<?=get_template_directory_uri();?>/images/social-vk-1.jpg"></a></center></div>
```

```
                    <div class="col-xs-3"><center><a
href="http://facebook.com/mpgu.edu" target="_blank"><img
src="<?=get_template_directory_uri();?>/images/social-fb-2.jpg"></a></center></div>
                    <div class="col-xs-3"><center><a
href="http://twitter.com/mpgu_official" target="_blank"><img
src="<?=get_template_directory_uri();?>/images/social-twitter-
3.jpg"></a></center></div>
                    <div class="col-xs-3"><center><a
href="http://www.youtube.com/user/mpguedu" target="_blank"><img
src="<?=get_template_directory_uri();?>/images/social-youtube-
4.jpg"></a></center></div>
                        </div>
                    </div>
                </div>
                <div class="row"></div>
    </div>
  <!-- / content -->
 <? get_footer(); ?>

<?php get_header(); $orderby = get_query_var('orderby'); ?>
<div class="container m30t">
        <div class="row">
                <div class="col-sm-12">
                        <?php the_breadcrumb(); ?>
                </div>
        </div>
<br>
        <div class="row title">
```

```
<div class="col-sm-6"><h2 class="books">Книжные материалы</h2></div>
<div class="col-sm-6">
    <div class="row">
        <div class="col-md-6 sort">Сортировать</div>
        <div class="col-md-6 sort">
            <a <?php if ($orderby == 'date') echo 'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'date')) ?>">Дате</a>
            <a <?php if ($orderby == 'comment_count') echo 'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'comment_count')) ?>">Популярности</a>
            <a <?php if ($orderby == 'modified') echo 'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'modified')) ?>">Новизне</a>
        </div>
    </div>
</div>
<div class="row books">
    <div class="col-sm-12"><div class="table-responsive">

    <?php
        $paged = (get_query_var('paged')) ? get_query_var('paged') : 1;

        $args = array (
            'cat'                => 17,
            'category__not_in'   => 14,
            'paged'              => $paged,
            'pagination'         => true,
```

```
                'posts_per_page'        => '10',
                'orderby'               => $orderby,
                'order'                         => 'ASC'
        );

        $wp_query = new WP_Query($args);
    ?>

    <table class="table">
        <thead><tr>
            <th width="2.5%"></th>
            <th width="38%">Наименование</th>
            <th width="25%">Автор</th>
            <th width="12%">Тип материала</th>
            <th width="10%">Год издания</th>
            <th width="10%">Кол-во страниц</th>
            <th width="2.5%"></th>
        </tr></thead>
        <tbody>

            <?php if ( $wp_query->have_posts() ) : while ( $wp_query->have_posts() ) : $wp_query->the_post();?>

                <tr onclick="location.href='<?php the_permalink(); ?>'">
                    <td class="addition"></td>
                    <td><?php the_title();?></td>
                    <td>
```

```
                    <?php $author = get_field('author'); ?>
                    <div style="padding: 17px 0 18px 0;" rel="books" data-content='<?= get_the_post_thumbnail($author[0]->ID, array(100, 100)); ?>'>
                    <?php
                        if ($author) echo $author[0]->post_title;
                        else echo 'Автор не указан.';
                    ?>
                    </div>
                </td>
                <td>
                    <?php
                        $m_mtype = get_field('m_mtype');
                        if (empty($m_mtype)) echo 'Тип не указан.';
                        else echo $m_mtype;
                    ?>
                </td>
                <td>
                    <?php
                        $m_year = get_field('m_year');
                        if (empty($m_year)) echo 'Год не указан.';
                        else echo $m_year;
                    ?>
                </td>
```

```
                                        <td>
                                            <?php
                                                $m_pages                    =
get_field('m_pages');
                                                if (empty($m_pages)) echo
'Год не указан.';
                                                else echo $m_pages;
                                            ?>
                                        </td>
                                        <td></td>
                                    </tr>
                                <?php endwhile; endif; ?>
                            </tbody>
                        </table>
                    </div></div>
                </div>

                <div class="row">
                    <div class="col-sm-12">
                        <?php
                            numeric_bootstrap_posts_nav($wp_query->max_num_pages);
                            wp_reset_postdata();
                        ?>
                    </div>
                </div>
            </div>

<?php get_footer(); ?>
```

```
<?php if (!is_user_logged_in()){header('location:/login/');exit;} get_header(); $orderby = get_query_var('orderby'); ?>
<div class="container m30t">
	<div class="row">
		<div class="col-xs-12">
			<?php the_breadcrumb(); ?>
		</div>
	</div>
	<div class="row m30t">
		<!-- <div class="col-xs-12 col-sm-4">
		<?php the_nav_menu(); ?>
	</div> -->
		<div class="col-xs-12 col-sm-12">
			<h1 class="deep-blue"><?php the_title();?></h1>
			<div class="row title">
		<div class="col-sm-6"><h2 class="books">Книжные материалы</h2></div>
		<div class="col-sm-6">
			<div class="row">
				<div class="col-md-6 sort">Сортировать</div>
				<div class="col-md-6 sort">
					<a <?php if ($orderby == 'date') echo 'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'date')) ?>">Дате</a>
					<a <?php if ($orderby == 'comment_count') echo 'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'comment_count')) ?>">Популярности</a>
					<a <?php if ($orderby == 'modified') echo 'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'modified')) ?>">Новизне</a>
```

```
                    </div>
                </div>
            </div>
        </div>
        <div class="row books">
            <div class="col-sm-12"><div class="table-responsive">

                <?php
                    $paged = (get_query_var('paged')) ? get_query_var('paged') : 1;

                    $current_user = wp_get_current_user();
                    $user_id = $current_user->ID;
                    $temp = get_user_meta( $current_user->ID, 'favorites' );
                    if ($temp == '') { $f = array(false); } else {
                        $f = explode(',', $temp[0]);
                    }

                    $args = array (
                        'cat'                   => 17,
                        'category__not_in'      => 14,
                        'paged'                 => $paged,
                        'pagination'            => true,
                        'post__in'              => $f,
                        'posts_per_page'        => '10',
                        'orderby'               => $orderby,
                        'order'                 => 'ASC'
                    );

                    $wp_query = new WP_Query($args);
```

```
?>

<table class="table">
    <thead><tr>
        <th width="2.5%"></th>
        <th width="38%">Наименование</th>
        <th width="25%">Автор</th>
        <th width="12%">Тип материала</th>
        <th width="10%">Год издания</th>
        <th width="10%">Кол-во страниц</th>
        <th width="2.5%"></th>
    </tr></thead>
    <tbody>

        <?php if ( $wp_query->have_posts() ) : while ( $wp_query->have_posts() ) : $wp_query->the_post();?>

            <tr onclick="location.href='<?php the_permalink(); ?>'">
                <td class="addition"></td>
                <td><?php the_title();?></td>
                <td>
                    <?php $author = get_field('author'); ?>
                    <div style="padding: 17px 0 18px 0;" rel="books" data-content='<?= get_the_post_thumbnail($author[0]->ID, array(100, 100)); ?>'>
                    <?php
```

```
                    if ($author) echo $author[0]->post_title;
                    else echo 'Автор не указан.';
                ?>
            </div>
        </td>
        <td>
            <?php
                $m_mtype = get_field('m_mtype');
                if (empty($m_mtype)) echo 'Тип не указан.';
                else echo $m_mtype;
            ?>
        </td>
        <td>
            <?php
                $m_year = get_field('m_year');
                if (empty($m_year)) echo 'Год не указан.';
                else echo $m_year;
            ?>
        </td>
        <td>
            <?php
                $m_pages = get_field('m_pages');
                if (empty($m_pages)) echo 'Год не указан.';
```

```
                                        else echo $m_pages;
                                    ?>
                                </td>
                                <td></td>
                            </tr>
                        <?php endwhile; endif; ?>
                    </tbody>
                </table>
            </div></div>
        </div>

        <div class="row">
            <div class="col-sm-12">
                <?php
                    numeric_bootstrap_posts_nav($wp_query->max_num_pages);
                    wp_reset_postdata();
                ?>
            </div>
        </div>
            </div>
    </div>
</div>

<?php get_footer(); ?>

<?php get_header(); ?>
<div class="container m30t">
        <div class="row">
            <div class="col-xs-12">
```

```
                <?php the_breadcrumb(); ?>
        </div>
</div>
<div class="row m30t">
        <div class="col-xs-12 col-sm-12">
                <h1 class="deep-blue">Import!!!</h1>
                <table>
                <? // сотрудник работает в организациях ?>

                <? // материал написан авторами ?>

                <? // материал из источников ?>
                <?
                        $paged = 1;
                        // WP_Query arguments
                        $args = array (
                                'post_type'             => 'post',
                                'posts_per_page'        => '1',
                                'paged'                         => $paged,
                                'cat'                                   => 'books',
                                //'meta_key'            => 'author_oldid',
                                //'order'               => 'ASC',
                                //'orderby'             => 'meta_value_num',
                        );

                        // The Query
                        $query = new WP_Query( $args );

                        // The Loop
```

```
                    if ( $query->have_posts() ) {
                        echo "Да будет экшн - ".$paged;
                        while ( $query->have_posts() ) {
                            $query->the_post();
                            $parentid = get_the_id();
                            //  update_field('field_5503f76433ff7',  array(),
$parentid);
                            // continue;
                            if (get_field('m_oldid_i')) {
                                $arr = explode(',', get_field('m_oldid_i'));
                                foreach ($arr as $key => $value) {
                                    $tempargs = array(
                                        'numberposts' => 1,
                                        'post_type' => 'sources',
                                        'meta_key' => 's_oldid',
                                        'meta_value' => $value,
                                        'compare' => '='
                                    );

                                    // get results
                                    $the_query = new WP_Query(
$tempargs );
                                    if( $the_query->have_posts() ){
                                        $temparr = array();
                                        while ( $the_query-
>have_posts() ) {
                                            $the_query->the_post();
                                            $newpostid =
get_the_id();
```

```
                                                $temparr[] = $newpostid;
                                                echo
'<tr><td>'.$parentid.' - </td><td>'.$value.' - </td><td>'.$newpostid.' - </td></tr>';
                                            }
                                            wp_reset_query();

    update_field('field_5506fd44f5345', $temparr, $parentid);
                                        }
                                    }
                                }
                            }
                        } else {
                            // no posts found
                            echo "Вот и все!";
                        }

                        // Restore original Post Data
                        wp_reset_postdata();
                    ?>

                    <?php $querystr = "SELECT DISTINCT meta_value FROM $wpdb-
>postmeta WHERE meta_key LIKE 'm_mtype' ORDER BY meta_value ASC";
                    $movie_names =  $wpdb->get_results($querystr, OBJECT);
                    echo sizeof($movie_names);
                    ?>
                    <ul>
                    <?php foreach ( $movie_names as $movie_name ){ ?>
                      <li><?php echo $movie_name->meta_value; ?> </li>
                    <?php } ?>
```

```
</ul>
</table>

<script type="text/javascript">



<?
$args = array (
        'post_type'             => 'authors',
        'posts_per_page'        => '10'
);

// The Query
$query = new WP_Query( $args );

if ( $query->have_posts() ) {
        while ( $query->have_posts() ) {
                $query->the_post();
                ?>
                        var input="<?=get_the_title();?> ученый";


        $.getJSON("https://ajax.googleapis.com/ajax/services/search/images?callback=?"
, {
                                        q: input,
                                        v: '1.0'
                                }, function(data) {
                                        $("#r").append('<img            src="'          +
data.responseData.results[0].url + '">');
```

```
                                    });
                                <?
                            }
                        }

                        ?>
                        </script>

                        <div id="r"></div>
                </div>
    </div>
</div>

<?php get_footer(); ?>

<?php get_header(); ?>
<div class="container m30t">
        <div class="row">
                <div class="col-xs-12">
                        <?php the_breadcrumb(); ?>
                </div>
        </div>
        <div class="row m30t">
                <?php if ( have_posts() ) : while ( have_posts() ) : the_post(); ?>
        <!-- <div class="col-xs-12 col-sm-4">
                <?php the_nav_menu(); ?>
        </div> -->
                <div class="col-xs-12 col-sm-12">
```

```
                    <h1 class="deep-blue"><?php the_title();?></h1>
                    <?php the_content();?>
                    <ul>
                    <?php
wp_list_categories('orderby=id&show_count=1&use_desc_for_title=0&child_of=7&hide_empty=0&title_li=0'); ?>
                    </ul>
                </div>
                <?php endwhile; endif; ?>
    </div>
</div>

<?php get_footer(); ?>

<?php get_header(); ?>
<div class="container m30t">
        <div class="row">
                <div class="col-xs-12">
                        <?php the_breadcrumb(); ?>
                </div>
        </div>
        <div class="row m30t">
                <?php if ( have_posts() ) : while ( have_posts() ) : the_post(); ?>
        <!-- <div class="col-xs-12 col-sm-4">
                <?php the_nav_menu(); ?>
        </div> -->
                <div class="col-xs-12 col-sm-12">
                        <?php
                                if (is_user_logged_in()){
```

```
                                ?>
                                    <h1 class="deep-blue">Личный кабинет</h1>
                                    <b>Вы подписаны на рассылку информации о новых материалах</b><br><button class="btn">Отписаться</button>
                                    <br>
                                    <br>
                                    <br>
                                <?
                                    echo do_shortcode("[wppb-edit-profile]");
                            } else {
                                ?>
                                    <h1 class="deep-blue"><?php the_title();?></h1>
                                <?
                                    the_content();
                            }
                            ?>
                    </div>
                    <?php endwhile; endif; ?>
    </div>
</div>

<?php get_footer(); ?>

<?php get_header(); ?>
<div class="container m30t">
        <div class="row">
                <div class="col-xs-12">
                        <?php the_breadcrumb(); ?>
                </div>
```

```
</div>
<div class="row m30t books">
        <div class="col-xs-12">
                <h1 class="deep-blue">Организации</h1>

                <table class="table padtable">
                <thead>
                        <th>Наименование организации</th>
                        <th>Сотрудников</th>
                </thead>
                        <tbody>
                        <?
                        $paged = (get_query_var('paged')) ? get_query_var('paged') : 1;
                                // WP_Query arguments
                        $args = array (
                                'post_type'             => 'organizations',
                                'pagination'            => true,
                                'paged'                                 => $paged,
                                'posts_per_page'        => '50',
                                'orderby'               => 'comment_count',
                        );

                        // The Query
                        $wp_query = new WP_Query( $args );

                        // The Loop
                        if ( $wp_query->have_posts() ) {
                                $pages = $wp_query->max_num_pages;
                                while ( $wp_query->have_posts() ) {
```

```
$wp_query->the_post();
// do something
?>
        <tr>
            <td><div style="padding: 17px 0 18px 0;"><a href="<? the_permalink();?>"><? the_title(); ?></a></td>
            <?
                $parent = get_the_id();

                // args
                $args = array(
                    'numberposts' => -1,
                    'post_type' => 'authors',
                    'meta_query' => array(
                        array(
                            'key' => 'author_orgs',
                            'value' => $parent,
                            'compare' => 'LIKE'
                        ),
                    )
                );

                $wp_query2 = new WP_Query($args);
                $q = $wp_query2->found_posts;
```

```
                                        ?>
                                        <td><? echo $q; ?></td>
                                    </tr>
                                <?
                            }
                        } else {
                            // no posts found
                        }

                        // Restore original Post Data

                        ?>
                        </tbody>
                    </table>
                    <center>
                        <center>
                        <?php numeric_bootstrap_posts_nav($wp_query->max_num_pages); ?>
                        </center>
                        </center>
                </div>
    </div>
</div>

<?php get_footer(); ?>

<?php get_header(); ?>
<div class="container m30t">
        <div class="row">
```

```
                <div class="col-xs-12">
                    <?php the_breadcrumb(); ?>
                </div>
        </div>
        <div class="row m30t">
                <?php if ( have_posts() ) : while ( have_posts() ) : the_post(); ?>
        <div class="col-xs-12 col-sm-4">
                <?php the_nav_menu(); ?>
        </div>
                <div class="col-xs-12 col-sm-8">
                    <h1 class="deep-blue"><?php the_title();?></h1>
                    <?php the_content();?>
                </div>
                <?php endwhile; endif; ?>
    </div>
</div>

<?php get_footer(); ?>

<?php get_header(); ?>
<div class="container m30t">
        <div class="row">
                <div class="col-xs-12">
                    <?php the_breadcrumb(); ?>
                </div>
        </div>
        <div class="row m30t">
                <div class="col-xs-12">
                    <h1 class="deep-blue">Схемы / карты</h1>
```

```
                    <? echo do_shortcode('[ess_grid alias="schemswithfilter"]');?>
                    <center>
                        <?php numeric_bootstrap_posts_nav($wp_query->max_num_pages); ?>
                    </center>
                </div>
    </div>
</div>

<?php get_footer(); ?>

<?php get_header(); $orderby = get_query_var('orderby'); ?>
<div class="container m30t">
        <div class="row">
                <div class="col-xs-12">
                    <?php the_breadcrumb(); ?>
                </div>
        </div>
        <br>
        <div class="row title">
                <div class="col-sm-6"><h2 class="authors">База авторов</h2></div>
                <div class="col-sm-6">
                    <div class="row">
                        <div class="col-md-6 sort">Сортировать</div>
                        <div class="col-md-6 sort">
                            <a <?php if ($orderby == 'date') echo 'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'date')) ?>">Дате</a>
```

```
                        <a <?php if ($orderby == 'comment_count') echo 'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'comment_count')) ?>">Популярности</a>
                        <a <?php if ($orderby == 'modified') echo 'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'modified')) ?>">Новизне</a>
                    </div>
                </div>
            </div>
        </div>

        <ul class="row authors">
            <?php
                $args = array (
                    'post_type'             => 'authors',
                    'pagination'            => true,
                    'paged'                 => $paged,
                    'posts_per_page'        => '8',
                    'orderby'               => $orderby,
                    'order'                 => 'DESC'
                );

                $wp_query = new WP_Query($args);
                if ( $wp_query->have_posts() ) : while ( $wp_query->have_posts() ) : $wp_query->the_post();
            ?>

            <li class="col-md-3">
```

```
                    <div rel="authors" data-content="<div class='popover-text'>полное умозрение строения и вождения кораблей, в пользу учащихся навигации
                                                                    <a href='#' class='details'>Подробнее</a>
                                                                    </div>">

                        <div class="name"><a href="<? the_permalink(); ?>"><?php the_title(); ?></a></div>
                        <div class="authors-thumbnail"><a href="<? the_permalink(); ?>"><?php the_post_thumbnail(array(142, 200)); ?></a></div>
                        <div class="time"><?php the_time('d.m.Y'); ?><a href="#"><span class="heart gray-heart"></span></a></div>

                    </div>
                </li>
                <?php endwhile; endif;?>
        </ul>

        <?php
                numeric_bootstrap_posts_nav($wp_query->max_num_pages);
                wp_reset_postdata();
        ?>
</div>

<?php get_footer(); ?>

<?php
        get_header();
        $orderby = get_query_var('orderby');
```

```
?>

<div class="container">

    <div class="row">
        <div class="col-sm-12">
            <?php the_breadcrumb(); ?>
        </div>
    </div>

    <div class="row title">
        <div class="col-sm-6"><h2 class="books">Книжные материалы</h2></div>
        <div class="col-sm-6">
            <div class="row">
                <div class="col-md-6 sort">Сортировать</div>
                <div class="col-md-6 sort">
                    <a <?php if ($orderby == 'date') echo 'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'date')) ?>">Дате</a>
                    <a <?php if ($orderby == 'comment_count') echo 'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'comment_count')) ?>">Популярности</a>
                    <a <?php if ($orderby == 'modified') echo 'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'modified')) ?>">Новизне</a>
                </div>
            </div>
        </div>
    </div>
```

```
<div class="row books">
	<div class="col-sm-12"><div class="table-responsive">

		<?php
			$paged = (get_query_var('paged')) ? get_query_var('paged') : 1;

			$args = array (
				'cat'                    => 17,
				'category__not_in'       => 14,
				'paged'                  => $paged,
				'pagination'             => true,
				'posts_per_page'         => '10',
				'orderby'                => $orderby,
				'order'                  => 'ASC'
			);

			$wp_query = new WP_Query($args);
		?>

		<table class="table">
			<thead><tr>
				<th width="2.5%"></th>
				<th width="38%">Наименование</th>
				<th width="25%">Автор</th>
				<th width="12%">Тип материала</th>
				<th width="10%">Год издания</th>
				<th width="10%">Кол-во страниц</th>
				<th width="2.5%"></th>
```

```
</tr></thead>
<tbody>

<?php if ( $wp_query->have_posts() ) : while ( $wp_query->have_posts() ) : $wp_query->the_post();?>

<tr onclick="location.href='<?php the_permalink(); ?>'">
<td class="addition"></td>
<td><?php the_title();?></td>
<td>
<?php $author = get_field('author'); ?>
<div style="padding: 17px 0 18px 0;" rel="books" data-content='<? echo get_the_post_thumbnail($author[0]->ID, array(100, 100)); ?>'>
<?php
if ($author) echo $author[0]->post_title;
else echo 'Автор не указан.';
?>
</div>
</td>
<td>
<?php
$m_mtype = get_field('m_mtype');
if (empty($m_mtype)) echo 'Тип не указан.';
```

```
                    else echo $m_mtype;
                ?>
            </td>
            <td>
                <?php
                    $m_year = get_field('m_year');
                    if (empty($m_year)) echo 'Год не указан.';
                    else echo $m_year;
                ?>
            </td>
            <td>
                <?php
                    $m_pages = get_field('m_pages');
                    if (empty($m_pages)) echo 'Год не указан.';
                    else echo $m_pages;
                ?>
            </td>
            <td></td>
        </tr>
    <?php endwhile; endif; ?>
    </tbody>
</table>
</div></div>
</div>

<div class="row">
```

```
            <div class="col-sm-12">
                <?php
                    numeric_bootstrap_posts_nav($wp_query->max_num_pages);
                    wp_reset_postdata();
                ?>
            </div>
        </div>

        <div class="row title">
            <div class="col-sm-6"><h2 class="authors">База авторов</h2></div>
            <div class="col-sm-6">
                <div class="row">
                    <div class="col-md-6 sort">Сортировать</div>
                    <div class="col-md-6 sort">
                        <a <?php if ($orderby == 'date') echo 'class="active"'; ?>
href="<?= add_query_arg(array('orderby' => 'date')) ?>">Дате</a>
                        <a <?php if ($orderby == 'comment_count') echo
'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'comment_count'))
?>">Популярности</a>
                        <a <?php if ($orderby == 'modified') echo
'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'modified'))
?>">Новизне</a>
                    </div>
                </div>
            </div>
        </div>

        <ul class="row authors">
            <?php
```

```
$args = array (
        'post_type'                 => 'authors',
        'pagination'        => true,
        'paged'                         => $paged,
        'posts_per_page'    => '8',
        'orderby'           => $orderby,
        'order'                     => 'DESC'
);

$wp_query = new WP_Query($args);
if ( $wp_query->have_posts() ) : while ( $wp_query->have_posts() ) : $wp_query->the_post();
    ?>

    <li class="col-md-3">
        <div rel="authors" data-content="<div class='popover-text'><?php the_title(); ?><br><br><?php echo get_field('author_bday');?><br><br><?php echo get_field('author_directions');?>
                                        <a href='<? the_permalink(); ?>' class='details'>Подробнее</a>
                                    </div>">

        <div class="name"><a href="<? the_permalink(); ?>"><?php the_title(); ?></a></div>
        <div class="authors-thumbnail"><a href="<? the_permalink(); ?>"><?php the_post_thumbnail(array(142, 200)); ?></a></div>
        <div class="time"><?php the_time('d.m.Y'); ?><a href="#"><span class="heart gray-heart"></span></a></div>
```

```
                </div>
            </li>
            <?php endwhile; endif;?>
        </ul>

        <?php
            numeric_bootstrap_posts_nav($wp_query->max_num_pages);
            wp_reset_postdata();
        ?>

        <div class="row title">
            <div class="col-sm-6"><h2 class="videos">Видео материалы</h2></div>
            <div class="col-sm-6">
                <div class="row">
                    <div class="col-md-6 sort">Сортировать</div>
                    <div class="col-md-6 sort">
                        <a <?php if ($orderby == 'date') echo 'class="active"'; ?>
href="<?= add_query_arg(array('orderby' => 'date')) ?>">Дате</a>
                        <a <?php if ($orderby == 'comment_count') echo
'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'comment_count'))
?>">Популярности</a>
                        <a <?php if ($orderby == 'modified') echo
'class="active"'; ?> href="<?= add_query_arg(array('orderby' => 'modified'))
?>">Новизне</a>
                    </div>
                </div>
            </div>
        </div>
```

```
<ul class="row videos">
    <?php
        $args = array (
            'cat'                   => 14,
            'pagination'            => true,
            'posts_per_page'        => '6',
            'orderby'               => $orderby,
            'order'                 => 'DESC'
        );

        $wp_query = new WP_Query($args);
        if ( $wp_query->have_posts() ) : while ( $wp_query->have_posts() ) : $wp_query->the_post();

            $content = get_the_content();
            preg_match_all('/[^?v=(.)]+$/', $content, $matches);

            $embed_code = wp_oembed_get('http://www.youtube.com/watch?v=' . $matches[0][0]);

            $category = get_the_category();
    ?>

    <li class="col-md-4">
        <div class="name"><?= get_the_title();?></div>
        <div class="adaptive-video"><?= $embed_code; ?></div>
        <div class="description">
            <span style="color: #535353;"><?php the_time('d.m.Y'); ?></span>
```

```
                        <?= $category[0]->cat_name; ?>
                    </div>

                </li>
                <?php endwhile; endif;?>
    </ul>

        <?php
            numeric_bootstrap_posts_nav($wp_query->max_num_pages);
            wp_reset_postdata();
        ?>
</div>

<?php get_footer(); ?>

<script type="text/javascript">
        $('[rel="books"]').popover({
            placement: 'left',
            html: 'true',
            animation: 'true',
            trigger: 'hover'
        });

        $('[rel="authors"]').popover({
            placement: 'top',
            html: 'true',
            animation: 'true',
            trigger: 'hover',
            delay: {show: 500, hide: 1000}
```

```
        });
</script>

<?php get_header(); ?>
<div class="container m30t">
        <div class="row">
                <div class="col-xs-12">
                        <?php the_breadcrumb(); ?>
                </div>
        </div>
        <div class="row m30t">
                <div class="col-xs-12">
                        <h1 class="deep-blue">Видео-записи</h1>

                        <? echo do_shortcode('[ess_grid alias="test"]');?>
                        <center>
                                <?php numeric_bootstrap_posts_nav($wp_query->max_num_pages); ?>
                        </center>
                </div>
    </div>
</div>

<?php get_footer(); ?>

<?php get_header(); $orderby = get_query_var('orderby'); ?>
<div class="container m30t">
        <div class="row">
                <div class="col-xs-12">
```

```
                    <?php the_breadcrumb(); ?>
                </div>
            </div>
            <div class="row m30t">
            <?php if ( have_posts() ) : while ( have_posts() ) : the_post(); ?>
                <h2><?php the_title();?></h2>
                <script type="text/javascript">
                    var input="<?=get_the_title();?> ученый";


            $.getJSON("https://ajax.googleapis.com/ajax/services/search/images?callback=?"
, {
                        q: input,
                        v: '1.0'
                    }, function(data) {
                        $("#aimg").append('<img src="' + data.responseData.results[0].url
+ '" class="img-responsive img-thumbnail" align="left" width="200" hspace="15">');

                    });
                </script>
                <div id="aimg"></div>
                <p><b>Годы жизни:</b> <?=get_field('author_bday');?>-
<?=get_field('author_dday');?></p>
                <p><b>Место рождения:</b> <?=get_field('author_bplace');?></p>
                <p><b>Направления деятельности:</b>
<?=get_field('author_directions');?></p>
                <?php the_post_thumbnail(array(142, 200)); ?>
                <?php echo get_the_content();?>
                <? if (get_field('author_externals')) {?>
```

```
<br>
<br>
    <h3>Внешние источники</h3>
    <ul>
        <?
        //echo get_field('author_externals');
        $externals = explode('; ', preg_replace('/\s\s+/', ' ', get_field('author_externals')) ); //explode(';', get_field('author_externals'));
        //print_r($externals);
        foreach ($externals as $value) {
            $temp = explode(": http", $value);
            echo "<li><a target='_blank' href='http".strip_tags($temp[1])."'>".strip_tags($temp[0]) ."</a></li>";
        }
        ?>
    </ul>
<? } ?>
<div class="clearfix"></div>
<? $aid = get_the_id(); ?>
<? if (get_field('author_orgs')) {
    $orgs = get_field('author_orgs');
    echo "<h3 class='m30t'>Автор работал в организациях</h3>";
    // WP_Query arguments
    echo "<ul>";
    foreach ($orgs as $post) {
        # code...
        setup_postdata($post);
        ?>
            <li>
```

```
                        <a href="<?php the_permalink(); ?>"><?php the_title(); ?></a>
                    </li>
                    <?
                }
                echo "</ul>";
                wp_reset_postdata();
            } ?>
            <h3 class="m30t">Материалы автора</h3>

        <?php endwhile; endif; ?>

            <div class="row books">
                <div class="col-sm-12"><div class="table-responsive">

                    <?php
                        $paged = (get_query_var('paged')) ? get_query_var('paged') : 1;

                        $args = array (
                            'post_type'         => 'post',
                            'paged'                                 => $paged,
                            'pagination'        => true,
                            'posts_per_page'    => '100',
                            'meta_query'        => array(
                                array(
                                    'key'       => 'author',
                                    'value'     => "" . $aid . "",
                                    'compare'   => 'LIKE',
```

```
                    ),
                ),
            );

            $wp_query = new WP_Query($args);
        ?>

        <table class="table">
            <thead><tr>
                <th width="5%"></th>
                <th width="44%">Наименование</th>
                <th width="17%">Тип материала</th>
                <th width="15%">Год издания</th>
                <th width="15%">Кол-во страниц</th>
                <th width="5%"></th>
            </tr></thead>
            <tbody>

                <?php if ( $wp_query->have_posts() ) : while ( $wp_query->have_posts() ) : $wp_query->the_post();?>

                    <tr onclick="location.href='<?php the_permalink(); ?>'">
                        <td class="addition"></td>
                        <td><?php the_title();?></td>
                        <td>
                            <div style="padding: 17px 0 18px 0;">
                                <?php
```

```
                                        $m_mtype =
get_field('m_mtype');
                                        if (empty($m_mtype))
echo 'Тип не указан.';
                                        else echo $m_mtype;
                                    ?>
                                    </div>
                                </td>
                                <td>
                                    <?php
                                        $m_year =
get_field('m_year');
                                        if (empty($m_year))
echo 'Год не указан.';
                                        else echo $m_year;
                                    ?>
                                </td>
                                <td>
                                    <?php
                                        $m_pages =
get_field('m_pages');
                                        if (empty($m_pages))
echo 'Год не указан.';
                                        else echo $m_pages;
                                    ?>
                                </td>
                                <td></td>
                            </tr>
                        <?php endwhile; endif; ?>
```

```
                                </tbody>
                            </table>
                        </div></div>
                </div>

                <div class="row">
                    <div class="col-sm-12">
                        <?php
                            numeric_bootstrap_posts_nav($wp_query->max_num_pages);
                            wp_reset_postdata();
                        ?>
                    </div>
                </div>

        <? get_template_part('widget', 'social-likes'); ?>

        <br>
        <?php comments_template(); ?>
        </div>
</div>

<?php get_footer(); ?>
```